



Architetture e tecnologie dei sistemi informatici

Argomenti (1/2)

- La struttura di un'applicazione software
- Le reti entro i sistemi informatici
- L'evoluzione tecnologica
- I servizi tipici
- Struttura dell'IT aziendale
- Il mondo del client-server
- Il problema delle compatibilità

Argomenti (2/2)

- Controllare i sistemi: TOGAF e ITIL
- Dispositivi di storage e architetture
- Panoramica sui sistemi operativi
- Panoramica sui linguaggi di sviluppo
- La Service Oriented Architecture (SOA)
- Cluster Computing
- Grid Computing
- Virtual computing e macchine virtuali
- Il Cloud Computing

Il sistema informatico

Abbiamo visto che un sistema informativo si suddivide in:

- **Risorse umane** (con organizzazione, ruoli, esperienze, ecc...)
 - **Risorse tecnologiche** (sistema informatico, inglese “IT System”)
 - **Risorse organizzative** (procedure, regolamenti, workflow, ecc...)
- Ma qual è la struttura del sistema informatico?

Programma software

- Applicazione software avente una sua identità precisa
- Un insieme di programmi può andare a formare un sistema software atto a svolgere le funzioni associate a uno o più processi business

Processo informatico

- E' un programma software in esecuzione
- Il processo informatico usa varie risorse (percentuale del tempo della CPU, memoria RAM, disco entro un computer e canali di comunicazione via rete) per svolgere il proprio compito

DBMS (Data Base Management System)

- E' un sistema software che standardizza l'accesso dei processi ai dati, offrendo delle interfacce generalizzate che permettono:
 - la condivisione dei dati da parte dei processi informatici
 - l'indipendenza dei dati rispetto ai processi
- Si definisce database o base di dati un insieme di archivi di dati gestiti contemporaneamente in modo efficiente ed unitario dal DBMS.

Scomposizioni dei Sistemi Informativi

- Su base funzionale
- Rispetto all'utenza
- Tecnologica/strutturale
- Per combinazione dei suddetti punti

La struttura di un'applicazione software

Categorie di applicazioni: ruoli

- **Interattive**: un operatore umano interagisce col programma
- **Macchina-macchina (M2M)**: due o più applicazioni comunicano tra loro
- **Batch**: un'applicazione inizia, legge dati, li elabora, li salva e poi termina

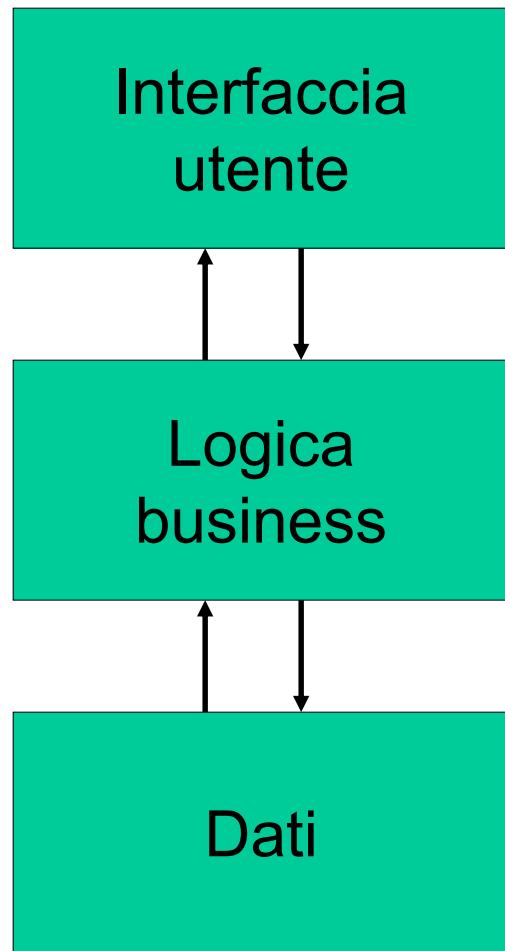
Categorie di applicazioni: architetture

- **Self-consistent**: un'applicazione completa come MS-Office
- **Client-server/strutturata**: applicazione suddivisa in moduli, che spesso lavorano su macchine diverse (es. il Web server e il browser)
- **Monolitica**: applicazione self-consistent non strutturata, che spesso svolge solo un semplice compito
- Basata su **(Micro)Servizi**: vedi più avanti

Struttura base di un'applicazione

- **Interfaccia utente** (grafica): presentazione dei dati e interazione con l'utente
- **Regole funzionali** (logica business): le procedure che compiono le operazioni in base ai comandi ricevuti dal livello precedente
- **Dati**: su cui si deve agire e che devono essere memorizzati (durano oltre i programmi)

La struttura dell'applicazione



Le reti entro i sistemi informatici

Sistema informatico e reti

- Il sistema informatico si compone di diverse risorse di calcolo connesse fra di loro in rete
- Quindi il sistema informatico è basato sulla rete
- La sua struttura si è evoluta nel tempo

Le reti in azienda

Il termine rete è nato per indicare in modo generico

- un collegamento tra due apparecchiature (sorgente e destinazione)
- attraverso un mezzo trasmittivo
- per effettuare una trasmissione di informazioni

Le reti in azienda: origini

- All'inizio le reti erano costituite essenzialmente da terminali remoti collegati a unità centrali (mainframe) mediante reti locali o linee telefoniche o telegrafiche
- l'uso di terminali remoti per l'elaborazione era noto come teleprocessing
- la potenza di elaborazione era concentrata in un punto (architettura centralizzata o master/slave)

Le reti in azienda: origini - 2

- La realizzazione di questo tipo di reti era legata a soluzioni proprietarie
- Una soluzione si dice proprietaria quando la realizzazione dipende dal costruttore ed è incompatibile con scelte di costruttori diversi
- Molto spesso le specifiche di una soluzione proprietaria non sono pubbliche

Le reti in azienda: oggi

- Attualmente per rete di calcolatori si intende un insieme di computer indipendenti,
- cioè che possono anche lavorare autonomamente
- ma collegati tra loro in modo da potersi scambiare informazioni (architettura distribuita)

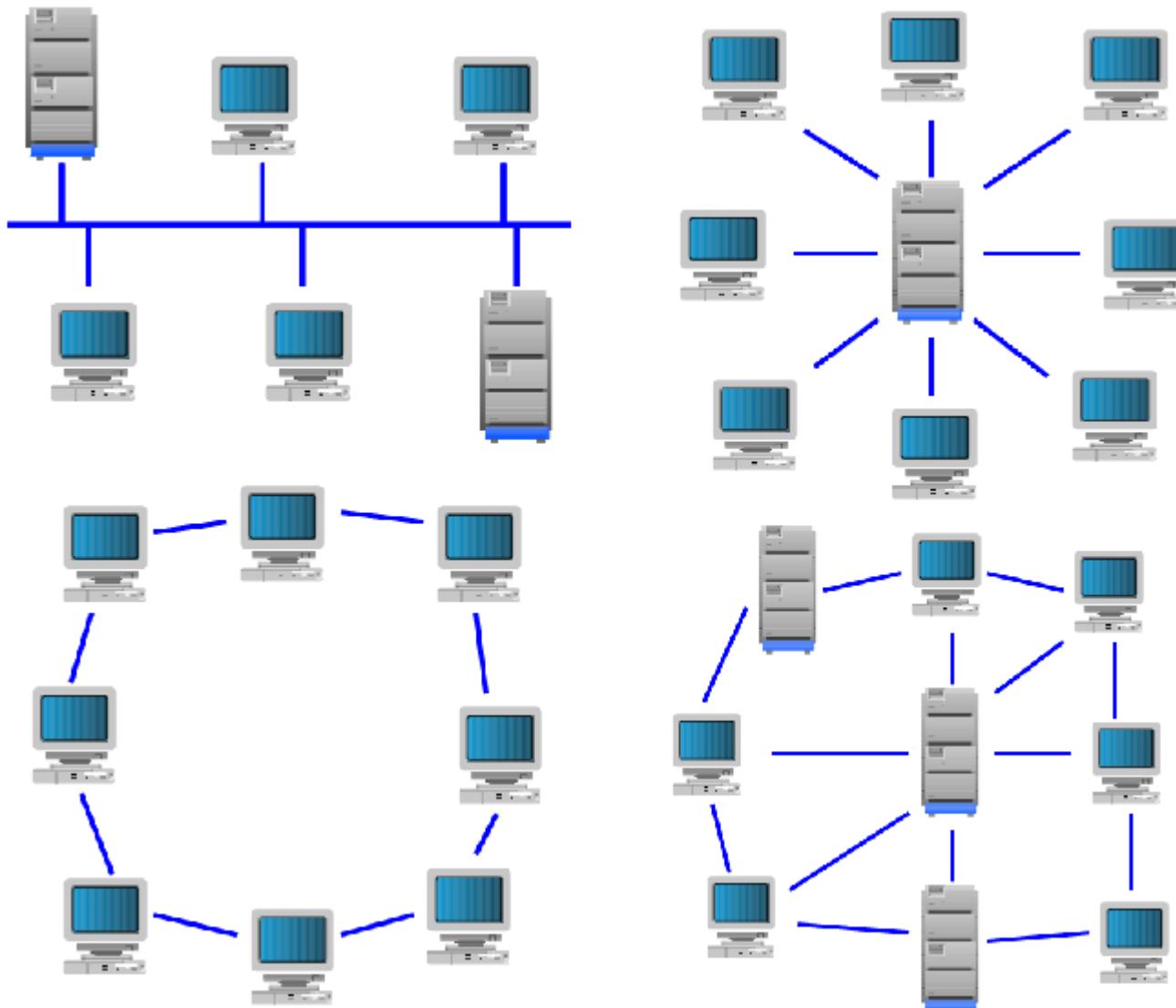
Le reti in azienda: oggi - 2

- Inoltre si è sentita la necessità di realizzare sistemi aperti che consentano di collegare e utilizzare prodotti di costruttori diversi;
- ciò rende necessario lo sviluppo di standard comuni (es. TCP/IP)
- un fattore importante è la scalabilità, cioè la possibilità di aumentare le risorse collegate alla rete in base alla necessità

Classificazione delle reti

- Local Area Network (LAN)
 - Metropolitan Area Network (MAN)
 - Wide Area Network (WAN)
-
- Intranet
 - Extranet
 - Internet

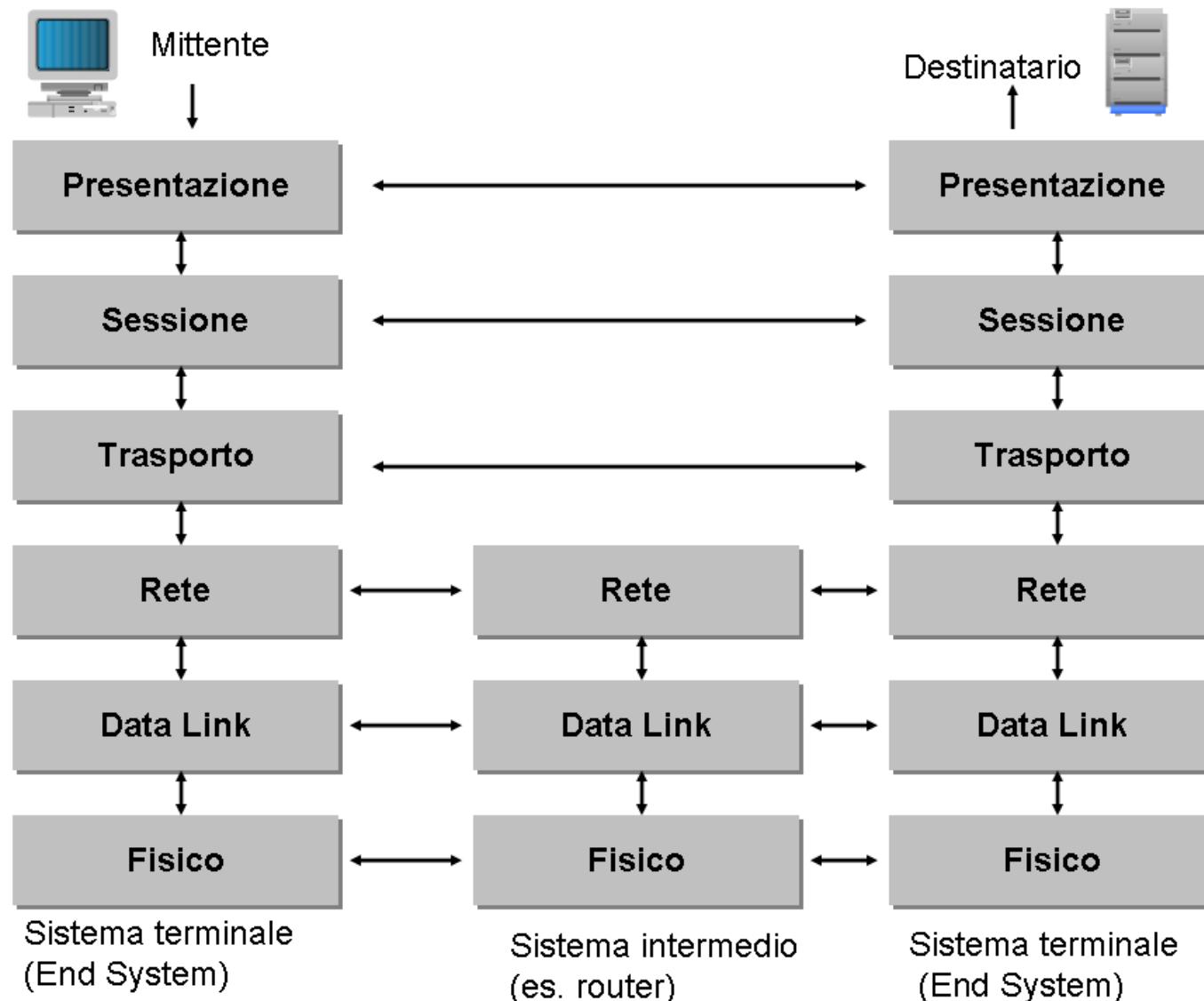
Topologia delle reti



Stratificazione delle reti: ISO/OSI (1/2)

- Applicazione
- Presentazione
- Sessione
- Trasporto (TCP)
- Rete (IP)
- Collegamento Dati (Ethernet, ADSL...)
- Fisico (cavo, wireless)

Stratificazione delle reti: ISO/OSI (2/2)



Esempi di standard a livello fisico

- Doppino telefonico
- Cavo coassiale
- Cavo a 4 coppie RJ-45 (UTP, FTP)
- Cavi in fibra ottica
- Sistemi wireless a corto raggio (Bluetooth, IRDA)
- Sistemi wireless LAN (Wi-Fi IEEE 802.11g)
- Sistemi wireless MAN/WAN (GPRS/EDGE/UMTS, Wi-MAX)
- Collegamento radio via satellite
- Collegamento ponte radio terrestre (es. HamRadio)
- Cavo elettrico

L'evoluzione tecnologica

Evoluzione dell'informatica

L'informatica nelle aziende ha assunto negli anni ruoli molto diversi:

Anni '70

- Il ruolo principale è quello di tecnologia di supporto alla produzione. Con obiettivi limitati e specifici.
- Tecnologie: Mainframe + terminali, elaboratori per automazione di processo.



Evoluzione dell'informatica

Anni '80

- Il ruolo principale è quello di tecnologia di supporto al coordinamento.
- Tecnologie: Personal computer, Reti aziendali proprietarie, risorse di I/O condivise.

Anni '90

- Il ruolo principale è quello di tecnologia di supporto alla intermediazione.
- Tecnologie: Internet, Intranet, Sistemi distribuiti, convergenza informatica-telecomunicazioni

Oggi

- Il ruolo principale è quello di tecnologia orientata ai servizi.
- Tecnologie: Architetture a livelli, Architetture Cloud, Peer-to-peer, Web services, Mobilità, ...

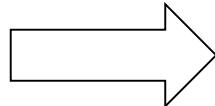
Outsourcing

- Per migliorare il rapporto costi/prestazioni, le aziende ricorrono all'**outsourcing** (gestione esterna) di elementi non essenziali della propria infrastruttura IT servendosi di Service Provider (SP).
- Possibili servizi gestiti esternamente:
 - Servizi fisici: web hosting, web portal, storage, ...
 - Servizi di manutenzione: apparati di rete, server, PC, applicativi, ...
 - Servizi globali: gestione globale dei servizi informatici.
- La gestione esterna dei servizi presuppone una diversa organizzazione interna e un diverso profilo professionale del personale interno.
- Occorre essere in grado di definire in modo molto preciso i compiti e le responsabilità di chi offre il servizio e le relative procedure di verifica delle prestazioni.

Outsourcing

- Sono presenti sul mercato aziende che offrono servizi sia a livello personale che a livello aziendale, con offerte ampie e articolate.
- Queste tipologie di infrastrutture richiedono sistemi con alti livelli di disponibilità e con capacità di adattamento dinamico alle mutevoli condizioni operative, in grado di allocare le risorse in modo efficiente, sicuro e conforme alle politiche stabilite fra i vari soggetti del sistema distribuito.
- Esempi:
 - Affitto server che offrono: servizi di rete, applicazioni specifiche, ...
 - Sistemi di uso generale: Google, MSN (My-Space, ..), Second-Life,

...

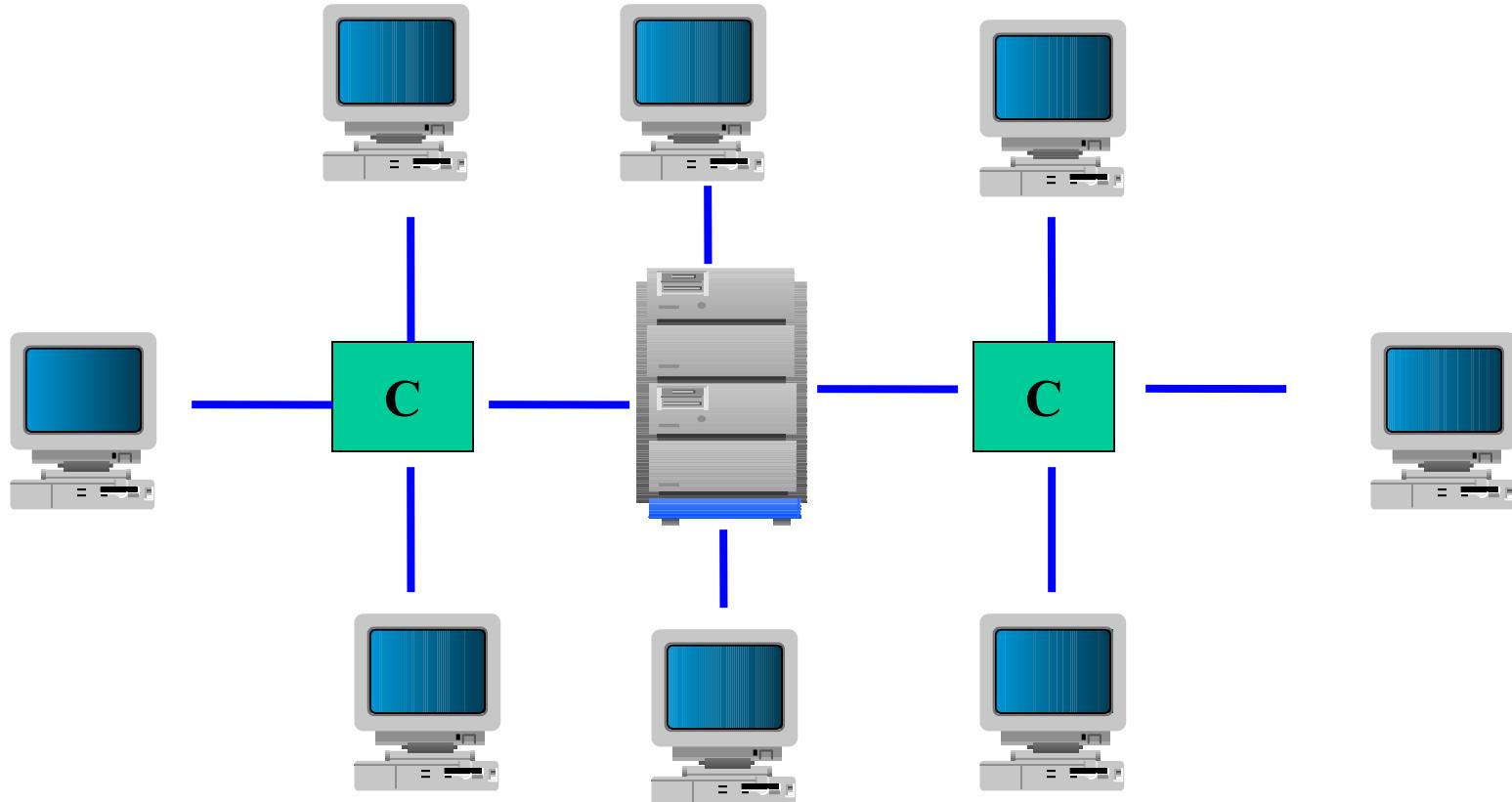


CLOUD COMPUTING

In principio era il Mainframe...

- Il modello “alla Mainframe” è tuttora molto diffuso
- Server centrale e terminali “dummy”
- Topologia di rete a stella

La rete a stella



Il vecchio paradigma

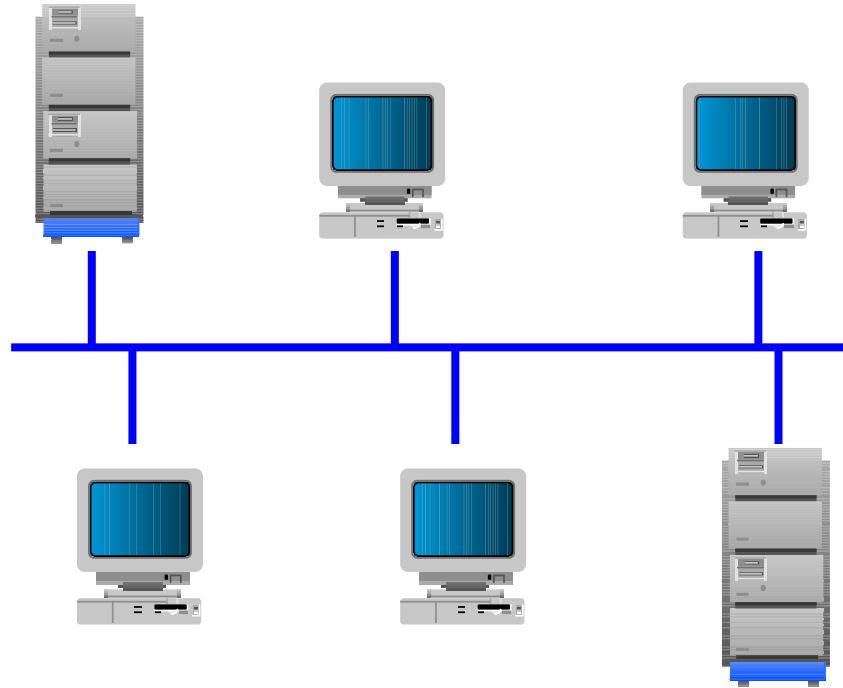


“Grandi” applicazioni entro sistemi operativi “piccoli”

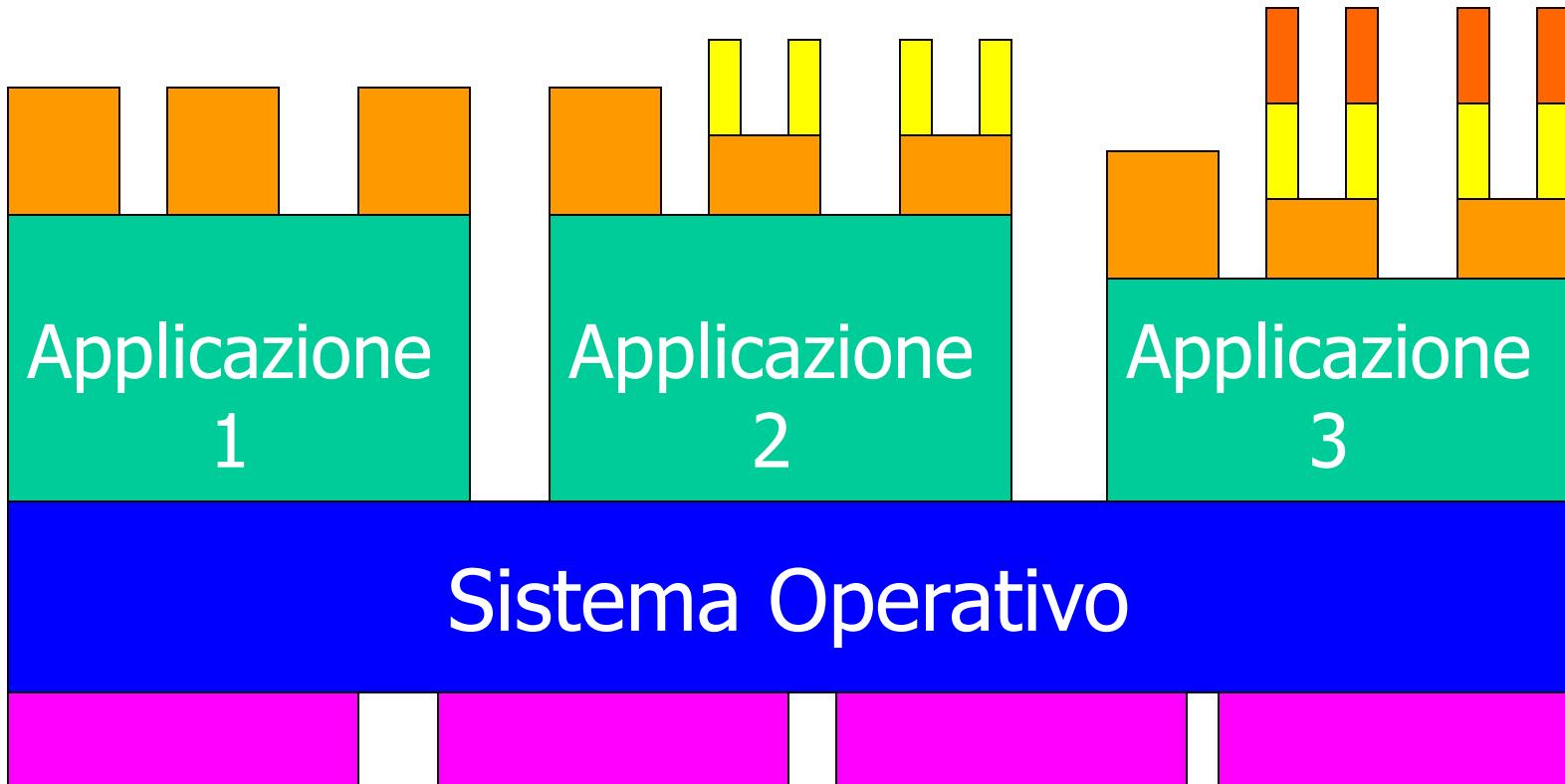
...e poi fu il Client-Server

- Introduzione di “intelligenza” nei Client
- Programmi sul client cooperano con programmi sul server

La rete a bus



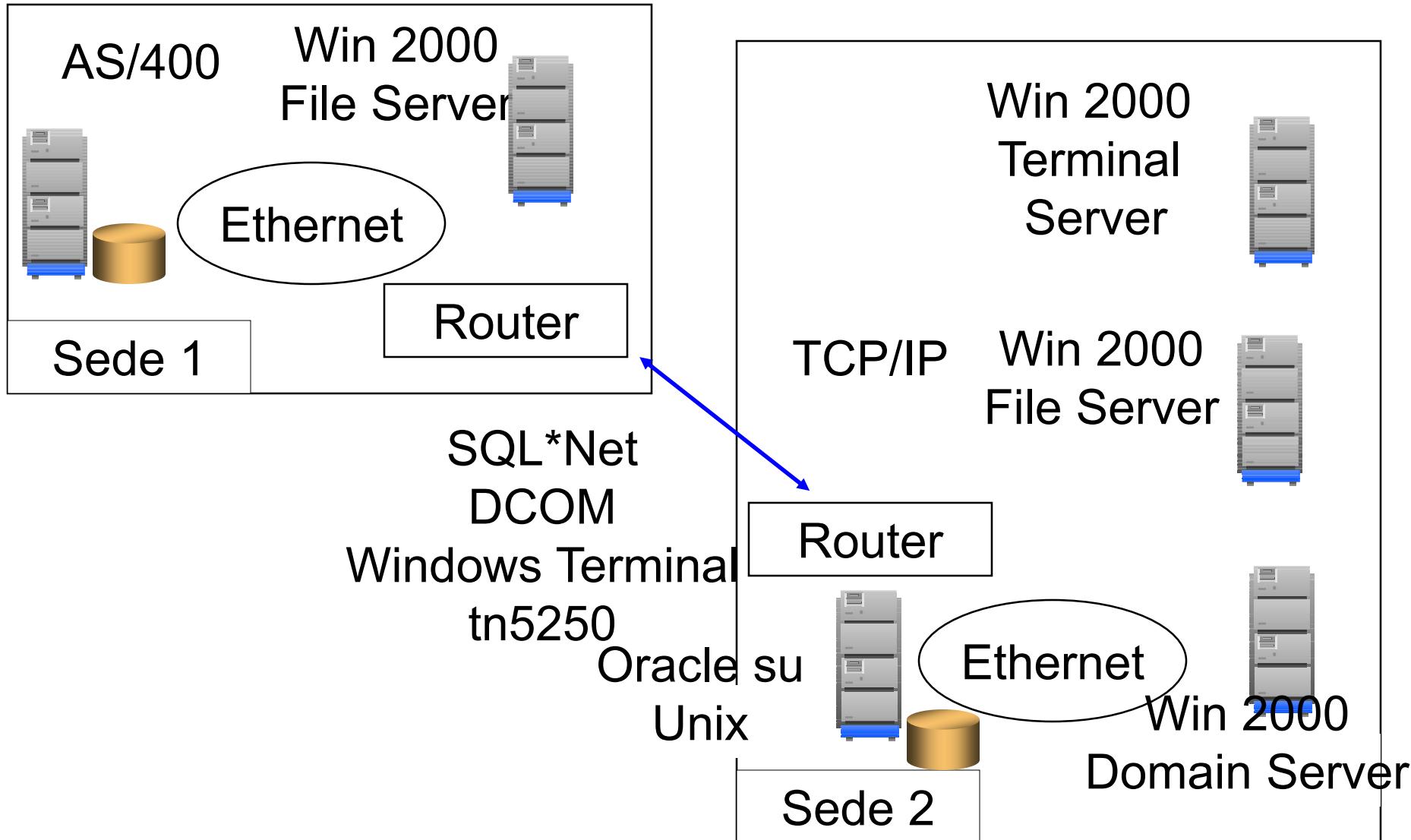
Il nuovo paradigma



Applicazioni “frammentate” in DLL, plug-in ecc...

Sistemi operativi: altrettanto
Presenza della rete

Un esempio di sistema informatico



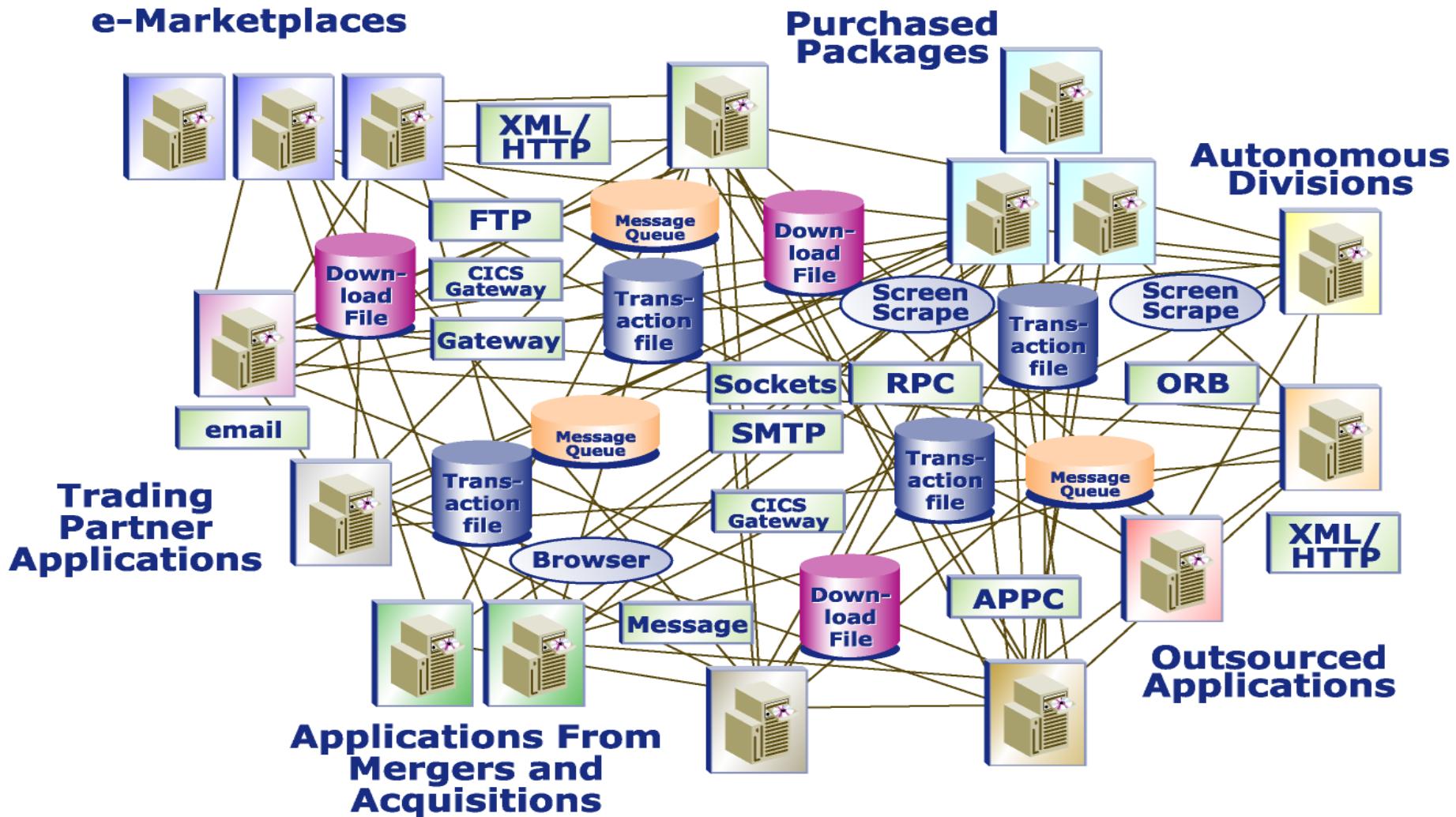
Il sistema informatico: evoluzione (1/2)

- Spesso nuovi applicativi
 - realizzati per rispondere alle mutate esigenze del business aziendale
-
- devono integrarsi con applicazioni ancora efficienti
 - la cui architettura è però ormai datata

Il sistema informatico: evoluzione (2/2)

- si ha quindi la convivenza di applicazioni
 - realizzate in epoche differenti
 - su piattaforme molto eterogenee
- Che devono collaborare (e quindi comunicare fra loro)

Rischio: spaghetti-integration

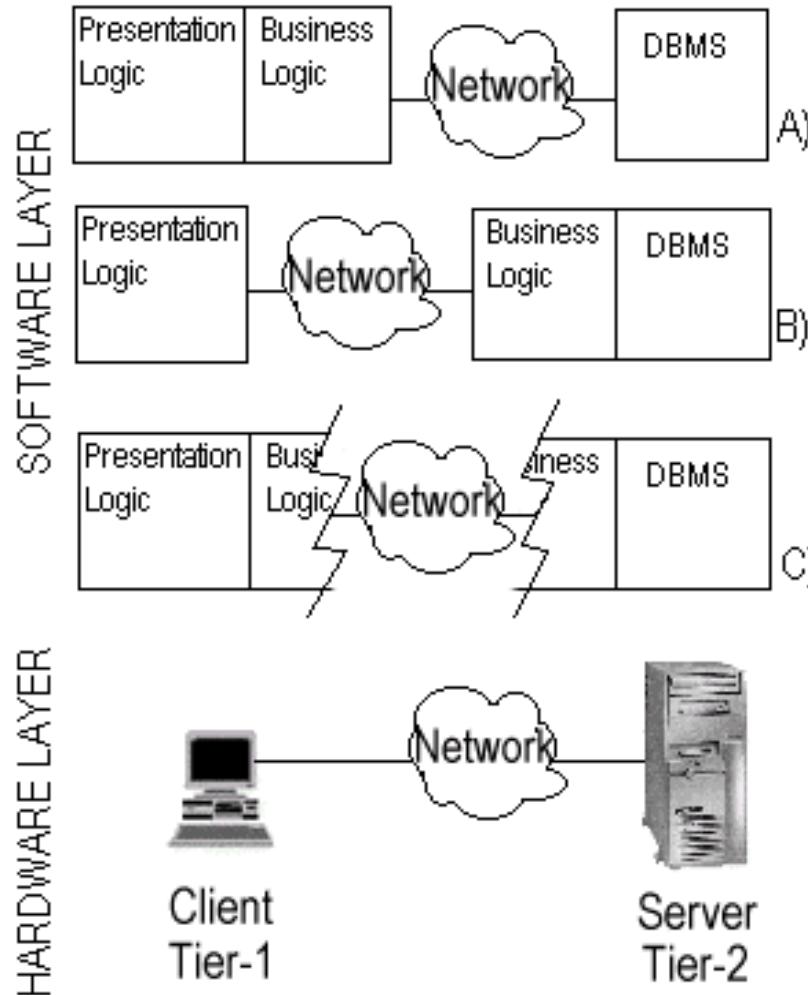


Il mondo del client-server

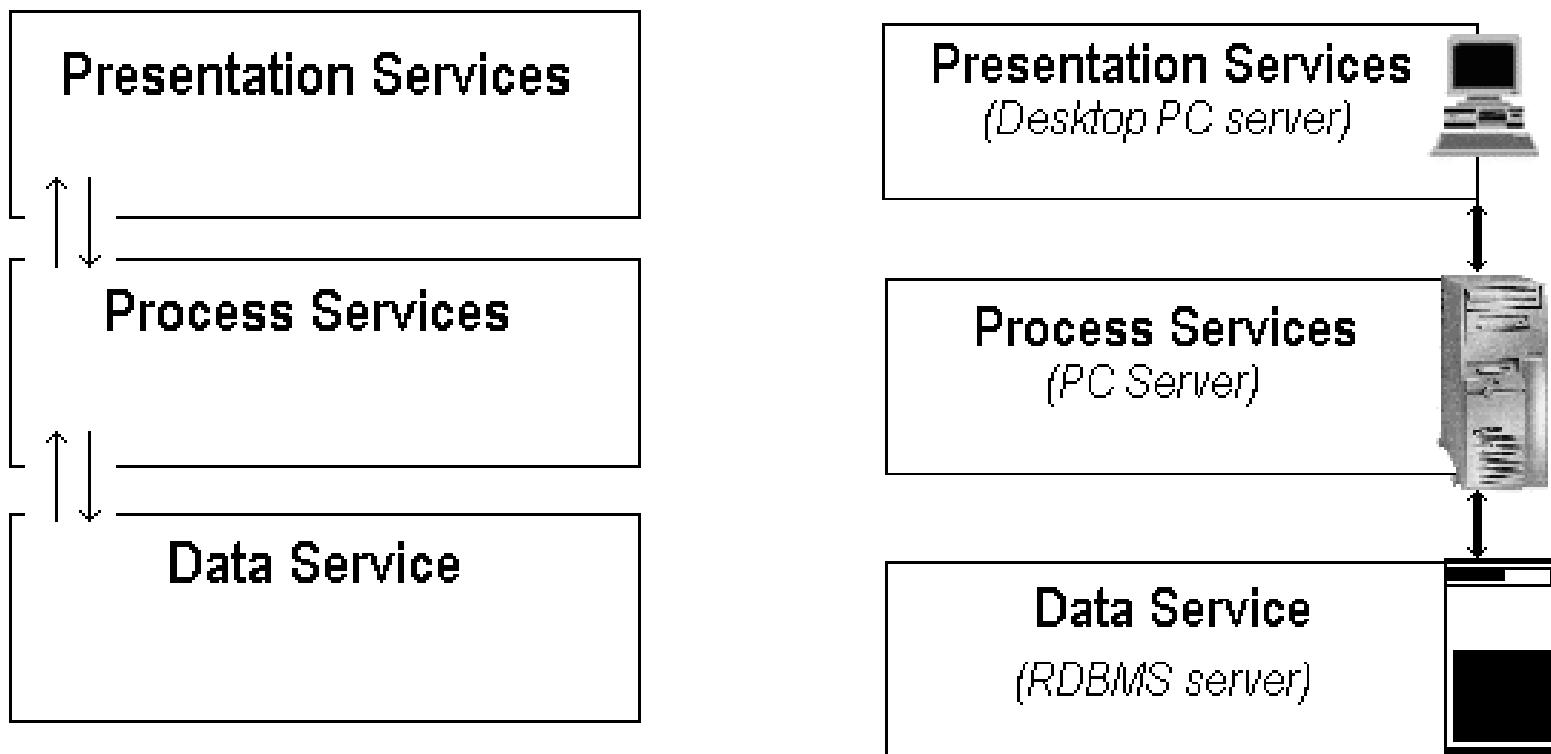
Il modello client-server

- Un **programma server** opera su un **computer server (host)**
- Un **programma client** opera su una **postazione client** (workstation)
- L'utente interagisce col programma client
- Il programma client dialoga col server **via rete**

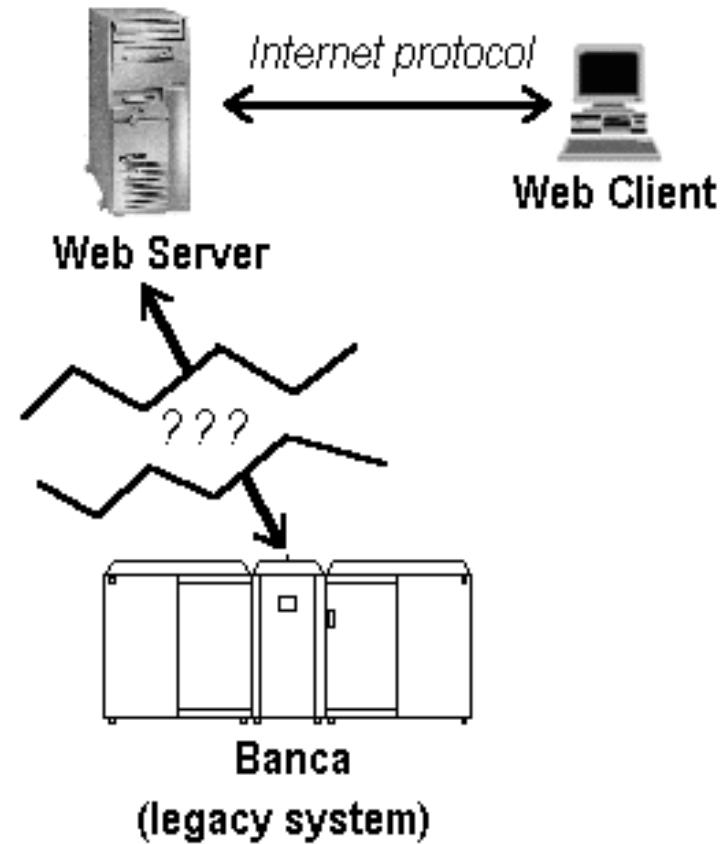
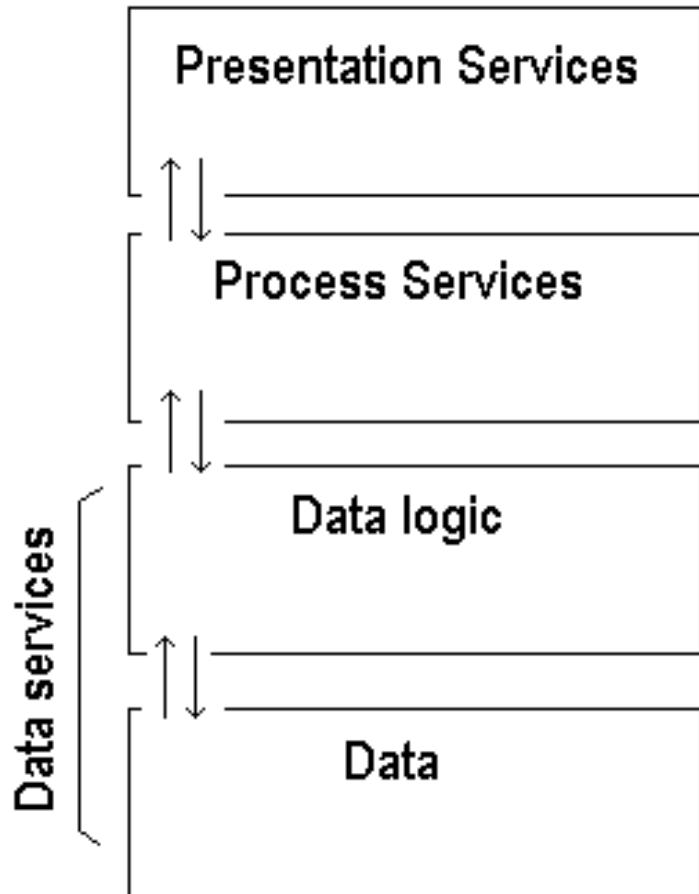
2-Tier Client/Server



3-Tier Client/Server

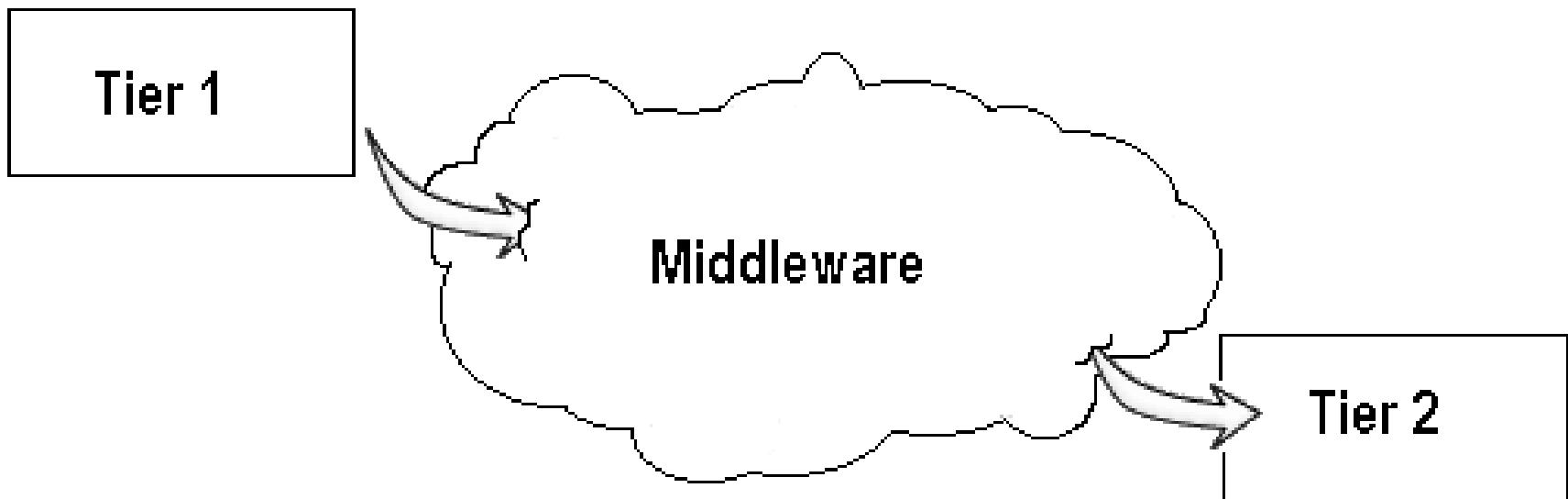


Multi-Tier Client/Server



Middleware

- Strato software che lega insieme diversi tier



Caratteristiche fondamentali del client-server

- Performance e scalabilità
- Complessità
- Supportabilità

I servizi tipici

Servizi standard di Internet

- Posta Elettronica (SMTP), porta 25
- Prelievo posta elettronica ricevuta (POP), porta 110
- Lettura posta elettronica in remoto (IMAP), porta 143
- Trasferimento file (FTP), porte 20 e 21
- World Wide Web (HTTP), porta 80
- Terminale remoto (Telnet), porta 23
- News (NNTP), porta 119

Servizi di Internet

- Domain Name System (DNS), porta 53
- SecureHTTP (HTTPS), porta 443
- Secure Shell e SecureFTP (SSH), porta 22
- Chat (ICQ), porta ?
- X-Window, porte 6000-6012

Protocolli proprietari su Internet

- Condivisione di dischi e stampanti
Microsoft/SAMBA (TcpBios), porte 137, 138, 139
- SQL*Net di Oracle, porta 1521
- ODBC di Microsoft, porta 1433
- Terminale remoto di Microsoft, porta 3389
- Iona Orbix (CORBA), porte 9000, 3085, 3094
- RealAudio e RealVideo

L'informatica “distribuita”

- File Server e Print Server
- Domain Server
- Database Server
- Groupware Server
- Web Server
- Transaction Server
- Object Server

File Server

- Servizio di Cartelle Condivise offerto da un server, centrale o dipartimentale
- Spazio fisicamente locato sul disco del server
- Privilegi di accesso stabiliti da tabella centralizzata o non di password e profili

File Server - 2

- Sistemi Apple
- Windows for Workgroup
- Windows .../7/10/11 & Server
- Unix-to-Unix (NFS)
- Unix-to-Windows (SAMBA)
- AS/400-to-Windows (Rumba e ClientAccess)

Print Server

- Servizio di stampa condivisa da più postazioni client
- Offerto da server centrale su cui risiedono anche i file temporanei della coda di stampa
- Non esente da problemi di safety e security

Domain Server

- Centralizzazione della gestione accessi e privilegi
- Mondo Windows: domain service e Active Directories
- Mondo Unix: Yellow Pages/NIS, NFS, OpenLDAP ed altro
- Sistemi ibridi: NFS+SAMBA Linux server

Administration/monitoring Server

Sorveglianza

- dei sistemi di domain
- dei server (es. processi, uso disco, uso memoria)
- della rete (sniffer, analizzatori di traffico)
- dei servizi (applicazioni, database)

Administration/monitoring Server: il mercato

- IBM Tivoli
- Computer Associates (CA) Unicenter
- Altri

Database Server

- Gestione centralizzata dei dati, condivisi fra tante postazioni client
- La logica funzionale può essere distribuita fra client e server

Il mercato dei Database Server

- Oracle Server
 - MS SQL Server
 - IBM DB2 e derivati di Informix
 - Sybase
 - CA Ingres
-
- MySQL
 - PostgreSQL

GroupWare Server

- Posta elettronica
- Raccolte integrate di posta e documenti vari (es. MS Exchange Server, Lotus Notes)
- Server di sviluppo per il lavoro in gruppo (es. CVS, RCCS, MS VisualSourceSafe, Git)

GroupWare Server: il mercato

- MS Exchange Server
 - IBM Lotus Notes
 - Novell OpenExchange
-
- CVS
 - RCCS
 - MS VisualStudio Team Edition

Transaction Server

- Server che ospita un sistema di coordinamento transazioni distribuite
- Es. IBM MQ Series (basato su code), Bea Tuxedo, MTS

Object Server

- Server che ospita un sistema per il funzionamento di oggetti distribuiti
- MS DCOM è garantito da Win2000
- CORBA
- Framework CORBA-based (es. BroadVision)
- Repository UDDI di SOAP (Web services)
- Sistemi Proprietari (es. Forté)

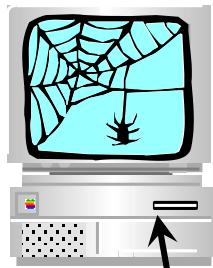
Web Server

I nodi Web o server Web rendono disponibili le informazioni in essi contenute sotto forma di **pagine ipertestuali**, contenenti **documenti multimediali** (ossia composti di testi, immagini fisse, filmati, suoni ecc...)

Il protocollo HTTP - 2

HTTP è un protocollo request/response

Web Browser
(HTTP client)

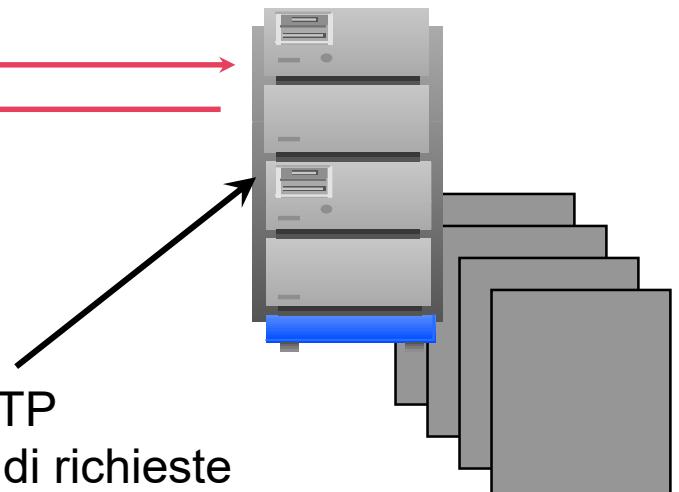


Il client HTTP costruisce ed invia le richieste quando viene specificata una URL

HTTP request

HTTP response

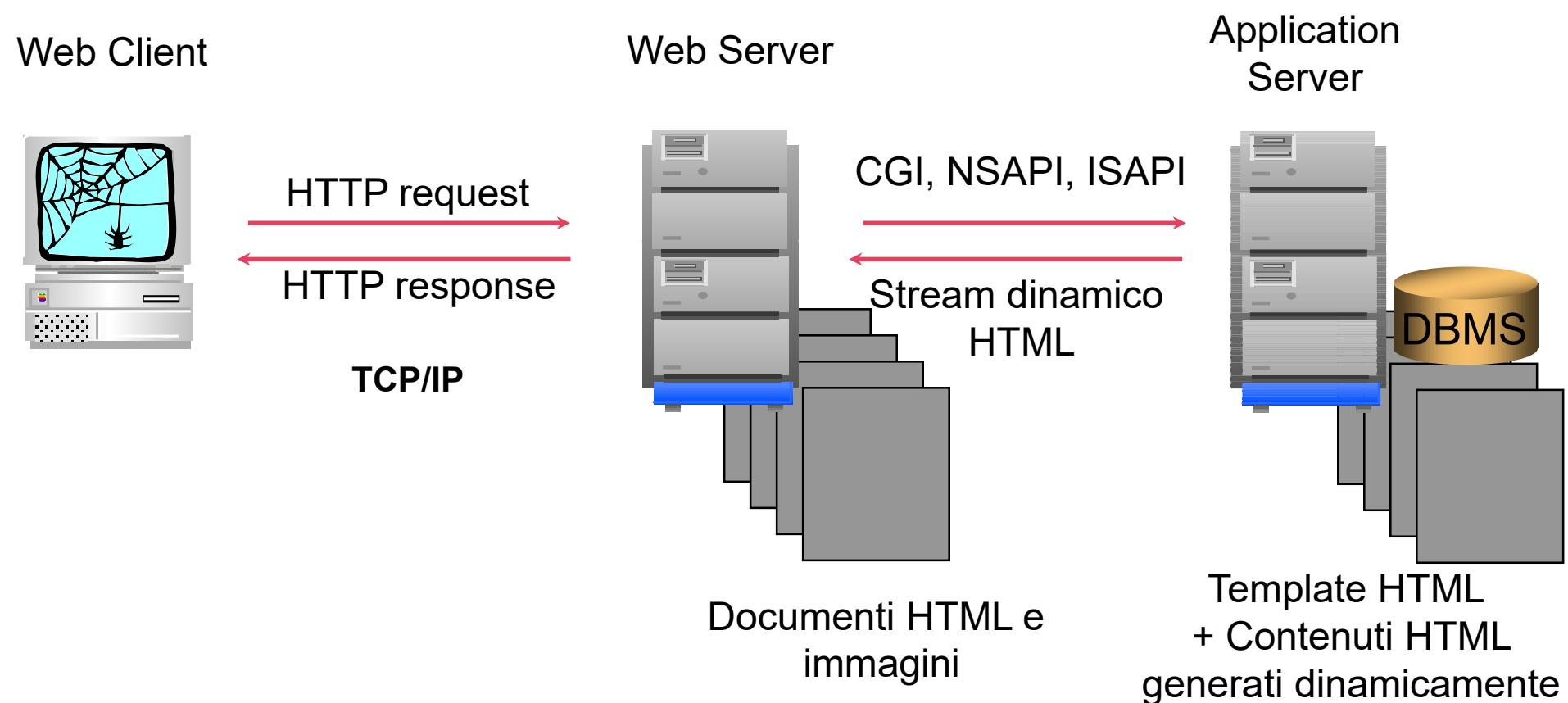
Web Server
(HTTP server)



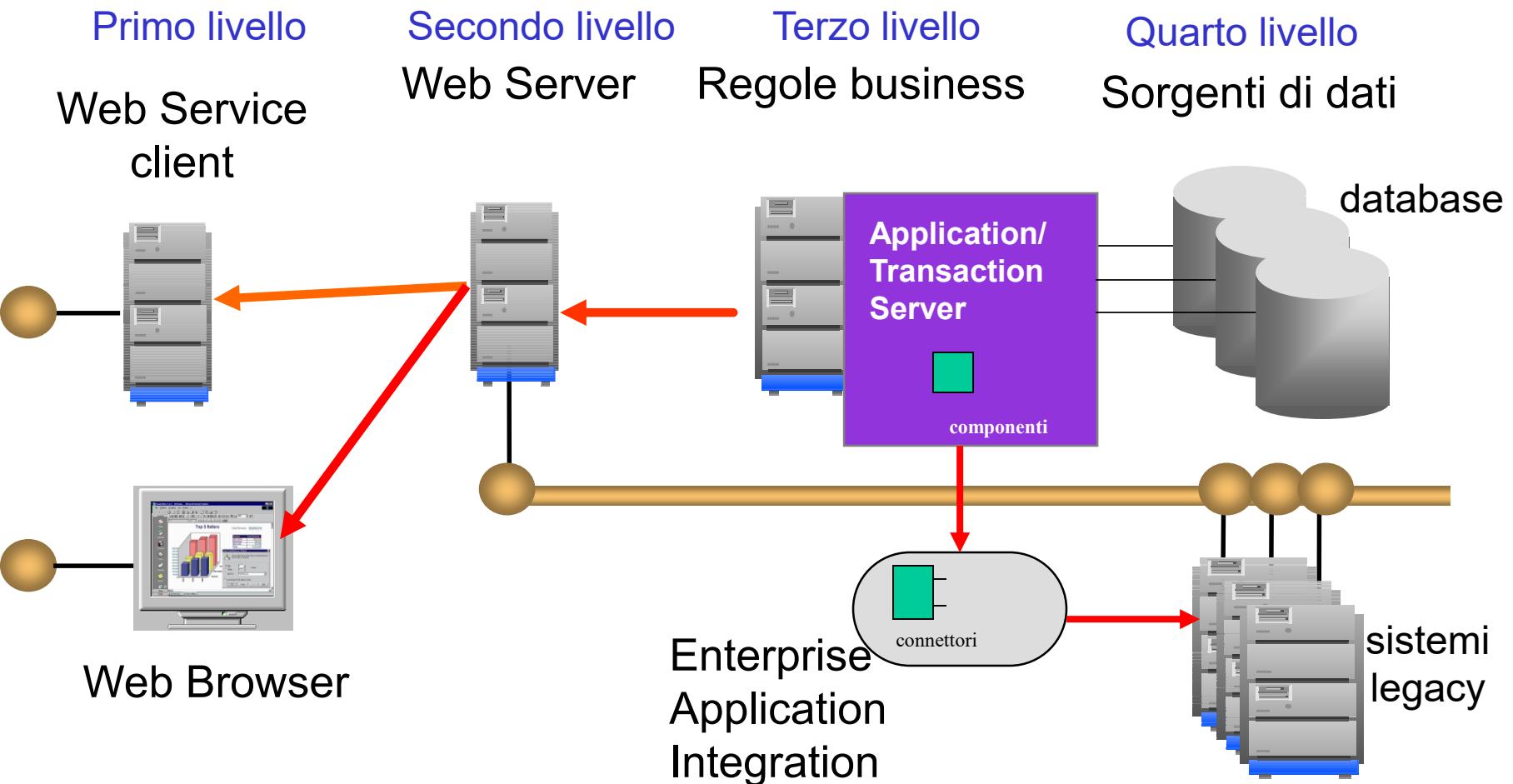
Il server HTTP
è in ascolto di richieste
HTTP sulla porta 80

Documenti HTML
+ Immagini

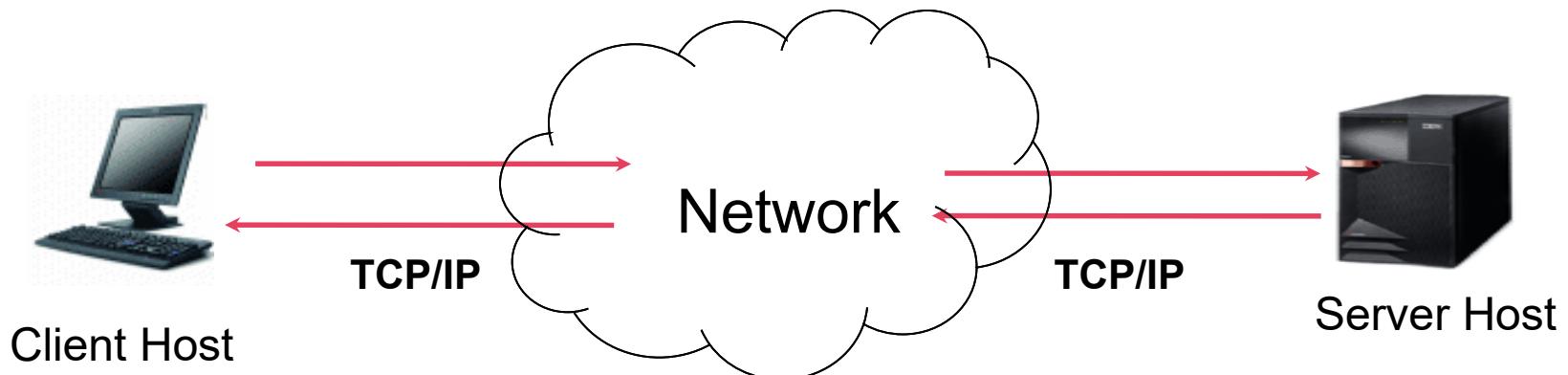
Stratificazione fisica di un sistema Web



Stratificazione fisica di un sistema Web - estensioni



Tipiche connessioni interattive su TCP/IP



- **telnet**: terminale a caratteri (UNIX, VMS)
- **tn3270/tn5250**: terminale a caratteri (MVS, OS/390, Z/OS, OS/400)
- **Windows Terminal**: terminale grafico/desktop remoto (Windows 2000/XP/2003)
- **PCAnywhere, VNC, CytrixWinFrame...**: terminale grafico/desktop remoto (Windows, Linux)
- **X-Window**: terminale grafico (UNIX)
- **Web Browser**: terminale HTML (tutti)

Struttura dell'IT aziendale

TOGAF/Archimate: lo schema dell'IT

Ambiente
Esterno

Prodotti /
Servizi

Livello
Business

Informazioni Processi Organizzazione

Livello
Applicazioni

Dati

Applicazioni

Livello
Tecnologia

Infrastruttura tecnologica

Strumenti IT nei sistemi informativi: scomposizione tecnologica

- Sistemi per l'utente finale (workstation client, Office suite, e-mail, applicazioni client)
- Applicazioni Enterprise (ERP, business intelligence, supporto ai progetti ecc...)
- Database (RDBMS, archivi come Lotus Notes etc...)

Strumenti IT nei sistemi informativi: infrastruttura tecnologica

- Sistemi Gestione Accessi/permessi (e.g. AD/LDAP, RACF ...)
- Sistemi operativi
- Hardware (Computer e altri dispositivi)
- Infrastrutture di rete
 - Software (librerie, configurazioni dei dispositivi ecc...)
 - Hardware (cablaggi, router, hub...)

Il problema delle compatibilità e le relazioni fra livelli di applicazioni

Scomposizione di architetture IT

Presentazione	Java AWT	Web Browser	MS Windows	Motif/X	3270/ 5250	Mac/ OS2				
Middleware	.NET COM/DCOM ActiveX	Corba	SOAP (HTTP/XML)	MQ - Series MSMQ						
Linguaggi	Web Scr.	(ERP) 4GL (e.g. ABAP)	Delphi	C/C++	COBOL	VB				
RDBMS	Ababas	IBM DB2 /UDB /400 /390	Informix	MS SQL Server	Oracle					
OS	Unix	NT	Midrange	Mainframe						
	HP UX	IBM AIX	Sun Solaris	Linux	Siemens	Compaq Digital	NT 2k/2003	OS/400	OS/390	
Hardware	PA	Power PC	Sparc		MIPS	Alpha	Alpha	Intel	AS/400	S/390

Le matrici di compatibilità: presentazione

Una determinata applicazione

- Possiede un look-and-feel dipendente dalle librerie grafiche usate
- Opera entro un window manager (es. Explorer di MS Windows o Gnome/Kde di X-Window)
- Se Web usa uno standard di HTML e JavaScript

Le matrici di compatibilità: presentazione

- I componenti grafici spesso dipendono dal linguaggio/ambiente di sviluppo
- Le librerie relative possono non essere presenti in tutti i sistemi operativi o nelle loro versioni
- I componenti web possono non essere compatibili con una certa versione di browser/Java/sistema ospite

Le matrici di compatibilità: comunicazione

- Il protocollo di comunicazione può richiedere librerie esterne all'applicativo
- Il middleware richiesto può non essere compatibile col sistema operativo
- Le comunicazioni interprocesso sfruttano meccanismi proprietari
- Il contesto di applicazione ha schermature (es. IM e firewall)

Le matrici di compatibilità: operazione

- Vengono richieste librerie run-time con una versione precisa
- Le librerie richieste non sono supportate dal sistema operativo o dalla configurazione presente
- Ciò può valere anche solo per una parte dell'applicativo (es. componenti Java, singole finestre VB)

Le matrici di compatibilità: logica business

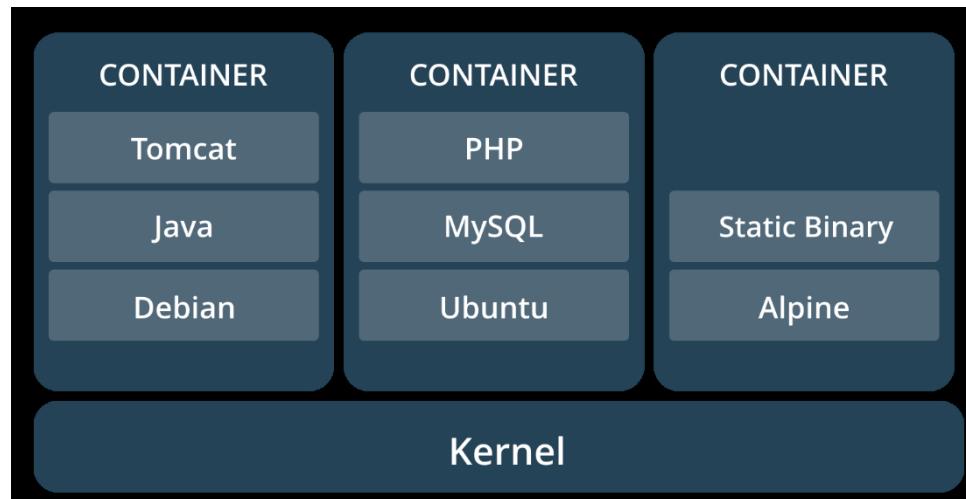
- L'applicativo è strutturato in componenti software e tutti devono essere presenti
- Vengono sfruttate librerie appartenenti ad altri pacchetti software

Le matrici di compatibilità: basi di dati

- L'applicativo sfrutta caratteristiche specifiche di un database (dialetti SQL, tipi di dati...)
- L'applicativo usa stored procedures
- L'applicativo richiede le librerie di connessione al database

Una soluzione al problema della compatibilità: i container

- Una immagine di container è un pacchetto leggero, isolato ed eseguibile di un software che include tutto ciò che è necessario per eseguirlo : codice, runtime, strumenti di sistema , librerie di sistema, configurazioni



- Disponibile per applicazioni Linux, Windows e Cloud, il software "containerizzato" funzionerà sempre allo stesso modo, indipendentemente dall'ambiente
- I contenitori isolano il software dall'ambiente circostante, ad esempio le differenze tra gli ambienti di sviluppo e di staging e aiutano a ridurre i conflitti tra i team che eseguono software diversi sulla stessa infrastruttura



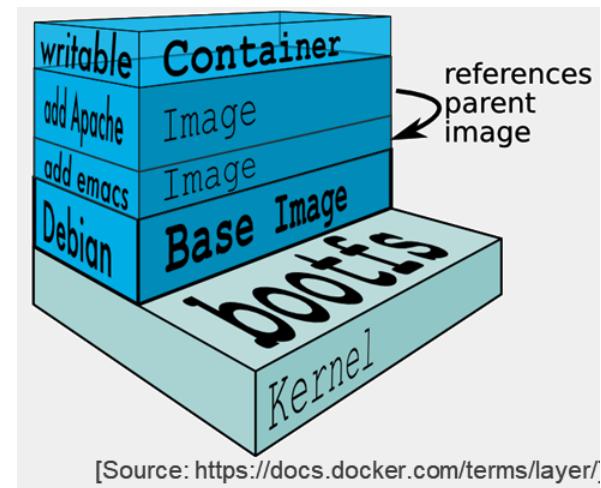
Docker is an open-source project that automates the deployment of applications inside software containers, by providing an additional layer of abstraction and automation of operating system-level virtualization on Linux.

[en.wikipedia.org]

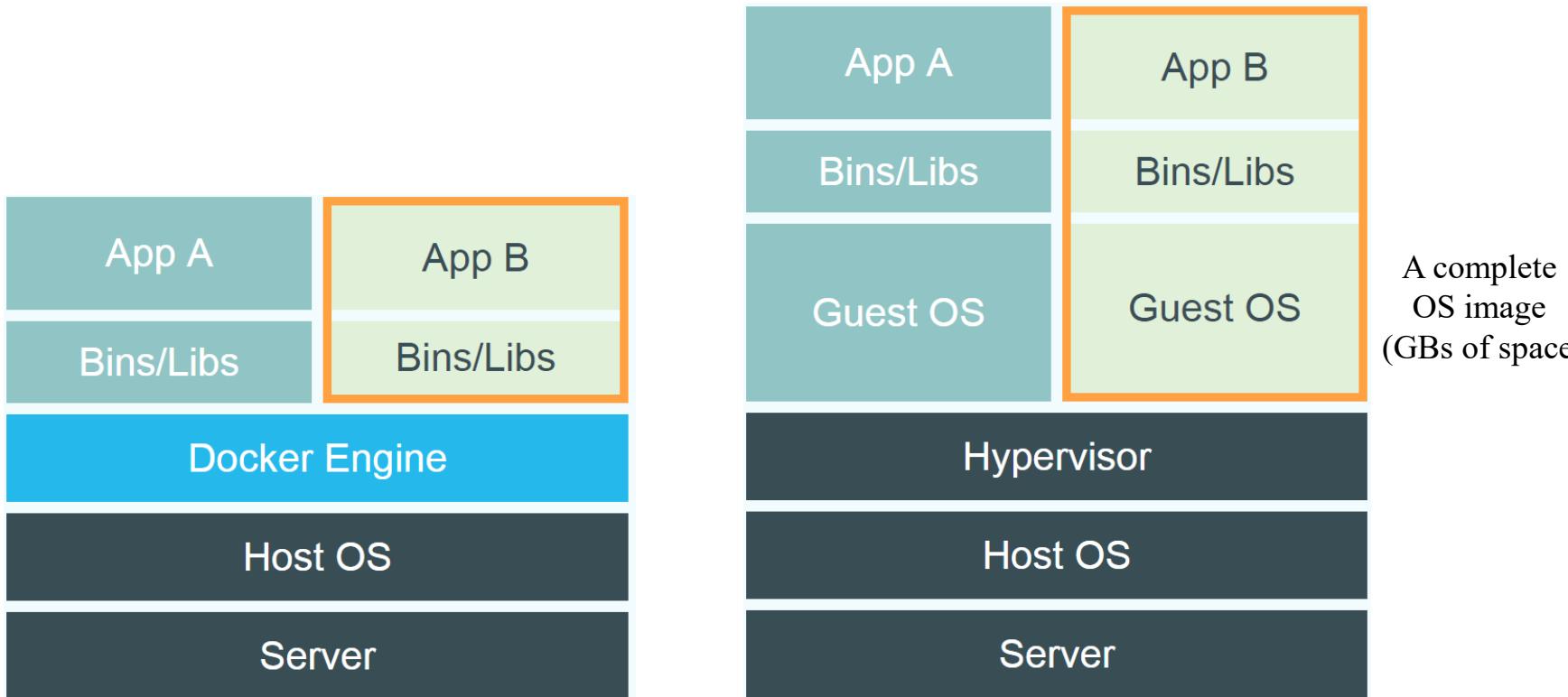
Docker technology

- libvirt: Platform Virtualization
- LXC (LinuX Containers): Multiple isolated Linux systems (containers) on a single host
- Layered File System

«Build, Ship and Run Any App, Anywhere»
[www.docker.com]



Docker vs Virtual Machine



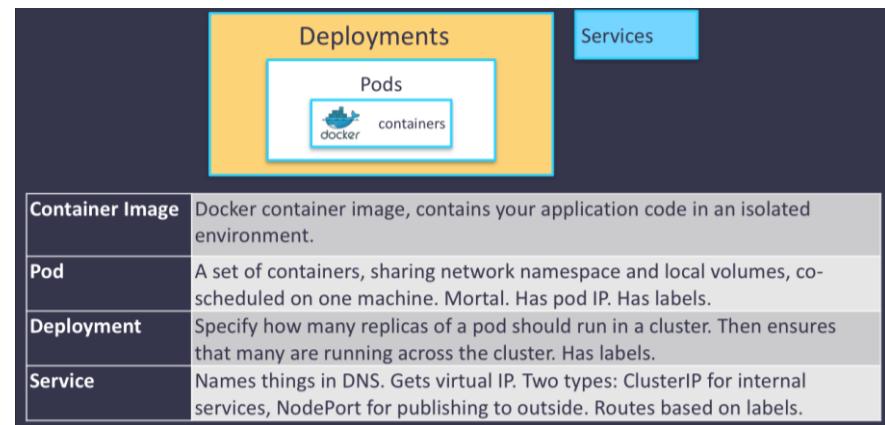
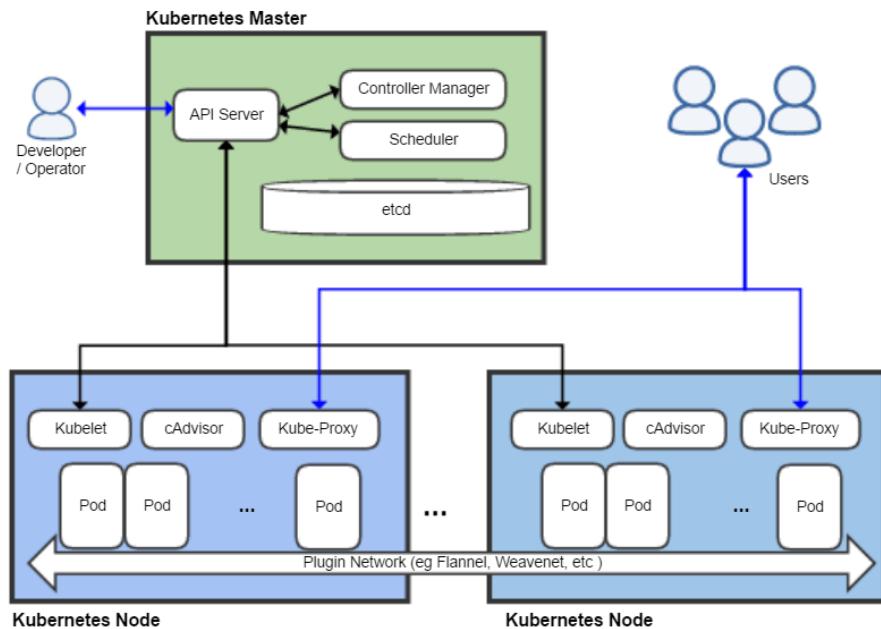
Multiple containers can run on the same machine and share the OS kernel with other containers, each running as isolated processes in user space. Containers take up less space than VMs (container images are typically tens of MBs in size), and start almost instantly.

Source: <https://www.docker.com/whatisdocker/>

Kubernetes (K8s)

- Un sistema open-source di orchestrazione e gestione di container
- Sviluppato da Google, ora da Cloud Native Computing Foundation
- Funziona con molti sistemi di containerizzazione, compreso Docker

Kubernetes è un termine greco per indicare il timoniere di una nave o un pilota



Le applicazioni moderne vengono sempre più spesso sviluppate con i contenitori, ovvero microservizi (vedi più avanti) inseriti in pacchetti insieme alle relative dipendenze e configurazioni

Kubernetes & DevOps

- Google ha progettato Kubernetes e poi lo ha rilasciato come open source per aiutare ad alleviare i problemi nei processi di DevOps
- L'obiettivo era quello di aiutare con l'automazione, la distribuzione e le metodologie agili per l'integrazione e la distribuzione del software
- Kubernetes ha reso più facile per gli sviluppatori passare dallo sviluppo alla produzione, rendendo le applicazioni più portatili e sfruttando l'orchestrazione
- Lo sviluppo in una piattaforma e il rilascio rapido, attraverso pipeline, in un'altra piattaforma ha mostrato un livello di portabilità che prima era difficile e ingombrante. Questo livello di astrazione ha contribuito ad accelerare DevOps e la distribuzione delle applicazioni

Che cos'è DevOps?

- DevOps riunisce i team tipicamente in silo diversi: **Sviluppo e Operazioni IT**
- DevOps promette di aiutare i team a lavorare collettivamente e in collaborazione per raggiungere più velocemente i risultati di business.
 - La sicurezza è anche una parte importante del mix che dovrebbe essere incluso come parte della cultura
 - Con **DevSecOps**, tre silo si uniscono come "cittadini di prima classe" che lavorano in collaborazione per ottenere lo stesso risultato
- Da un punto di vista tecnologico, DevOps si concentra tipicamente su CI/CD (Continuous Integration - Continuous Distribution).
 1. **Integrazione continua**: gli sviluppatori effettuano aggiornamenti costanti del codice sorgente all'interno di un repository condiviso, che viene poi scansionato e controllato da una build automatizzata, consentendo ai team di rilevare tempestivamente i problemi
 2. **Distribuzione continua**: una volta approvato, il codice viene rilasciato in produzione, con il risultato che ogni giorno vengono effettuate molte distribuzioni di produzione
 3. **Consegna continua**: il software viene costruito e può essere rilasciato in qualsiasi momento - ma con un processo manuale

Controllare i sistemi: ITIL e TOGAF

TOGAF/Archimate: lo schema dell'IT

Ambiente
Esterno

Prodotti /
Servizi

Livello
Business

Informazioni Processi Organizzazione

Livello
Applicazioni

Dati

Applicazioni

Livello
Tecnologia

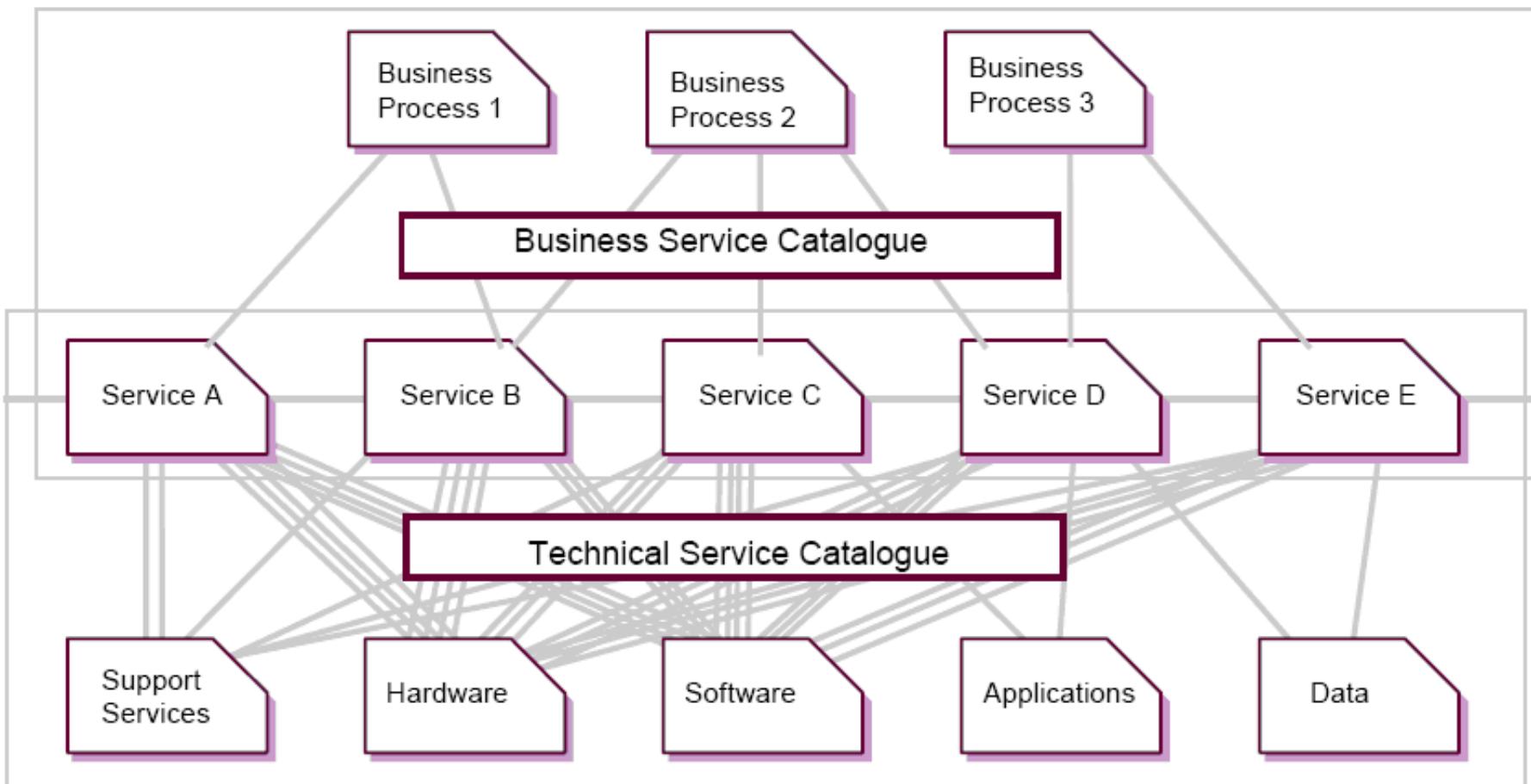
Infrastruttura tecnologica

Il Service Catalogue di ITIL

Il Service Catalogue ha due aspetti:

- Il **Business Service Catalogue** che contiene i dettagli di tutti i servizi IT erogati ai clienti, assieme alle relazioni con le unità e i processi di business che fanno affidamento sui servizi IT. E' la vista cliente del Service Catalogue
- Il **Technical Service Catalogue** contiene i dettagli di tutti i servizi IT erogati ai clienti, assieme alle relazioni con i servizi di supporto, i servizi condivisi, i componenti e i Configuration Item necessari alla fornitura del servizio al business. Dovrebbe sostenere il BSC ma non essere parte della vista cliente

ITIL Service Catalogue (1/2)



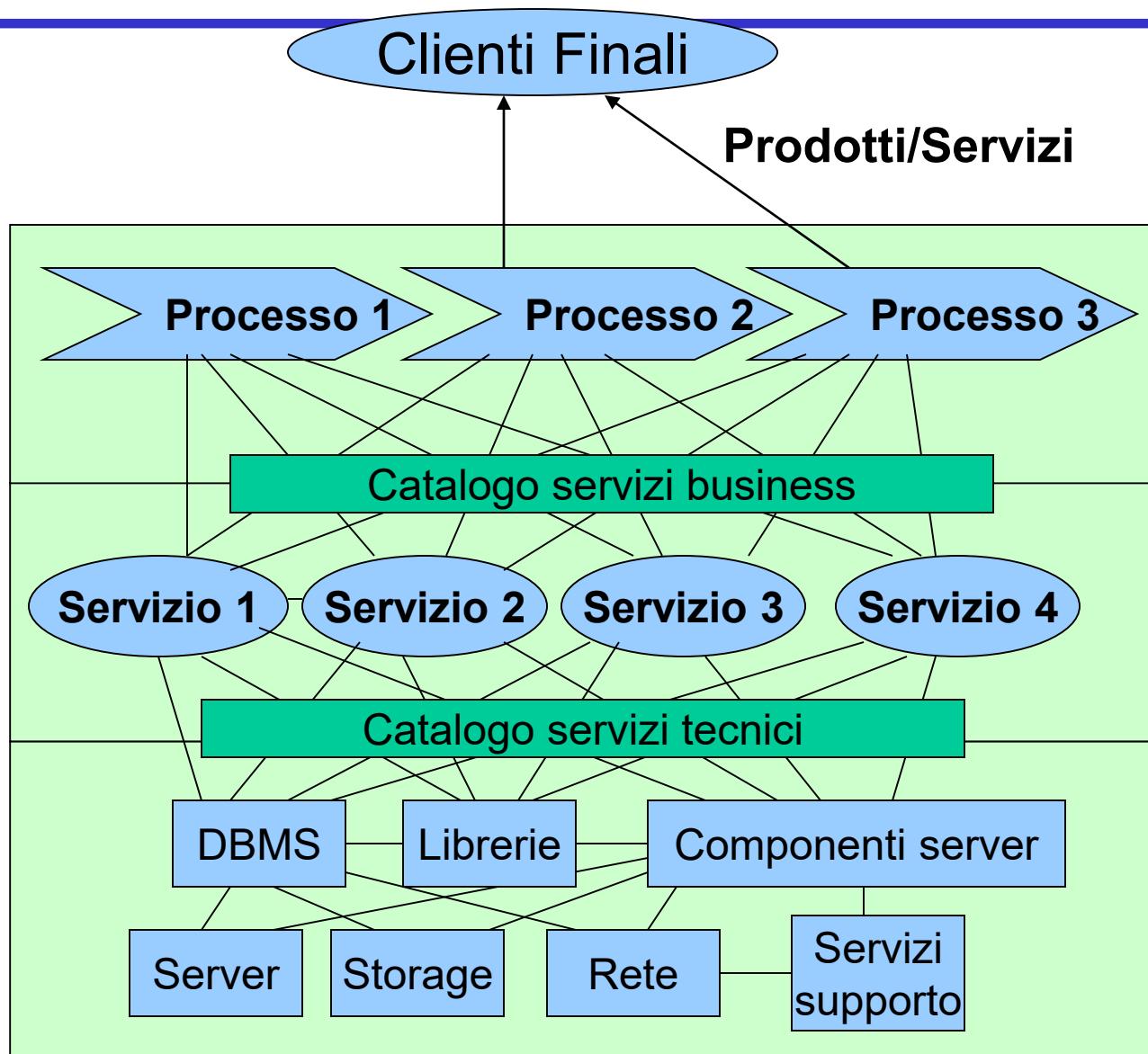
ITIL Service Catalogue (2/2)

Ambiente
Esterno

Livello
Business

Livello
Applicazioni

Livello
Tecnologia



Cataloghi dei servizi di business e IT

Business Service Catalogue (BSC)

- La vista strategica: il 'cosa': descrive il servizio a tutti gli utenti/clienti, le funzioni di business che supporta e i gruppi di utenti:
 - costo del servizio (agli utenti) e come è pagato
 - informazioni su SLA
 - gruppi di utenti che lo utilizzano
 - misure di qualità o KPI

Technical Service Catalogue (TSC)

- La vista implementativa - il 'come'. Per ogni servizio del BSC, si descrive come è realizzato e erogato:
 - Link e riferimenti ad altri servizi che sono essenziali per l'erogazione di questo servizio
 - Configuration Items (apparati, software, server, etc.)
 - Procedure di startup e di shutdown
 - Informazioni su ripristino e "piano B"
 - Principali contratti di supporto e punti di contatto/referenti

Configuration Item

- Nella terminologia del Configuration Management ITIL, i componenti IT ed i servizi con essi forniti sono noti come **Configuration Item** (CI).
- Tra i Configuration Item su cui si basa un servizio IT sono presenti i diversi elementi (servizi di supporto, servizi condivisi, componenti, etc.) su cui si basa

Configuration Item

I CI possono includere

- l'hardware dei PC,
- i vari tipi di software,
- i componenti di rete sia attivi che passivi,
- i server,
- i processori,
- la documentazione,
- le procedure,
- i servizi
- e tutti gli altri componenti IT che vanno controllati dall'Organizzazione IT.

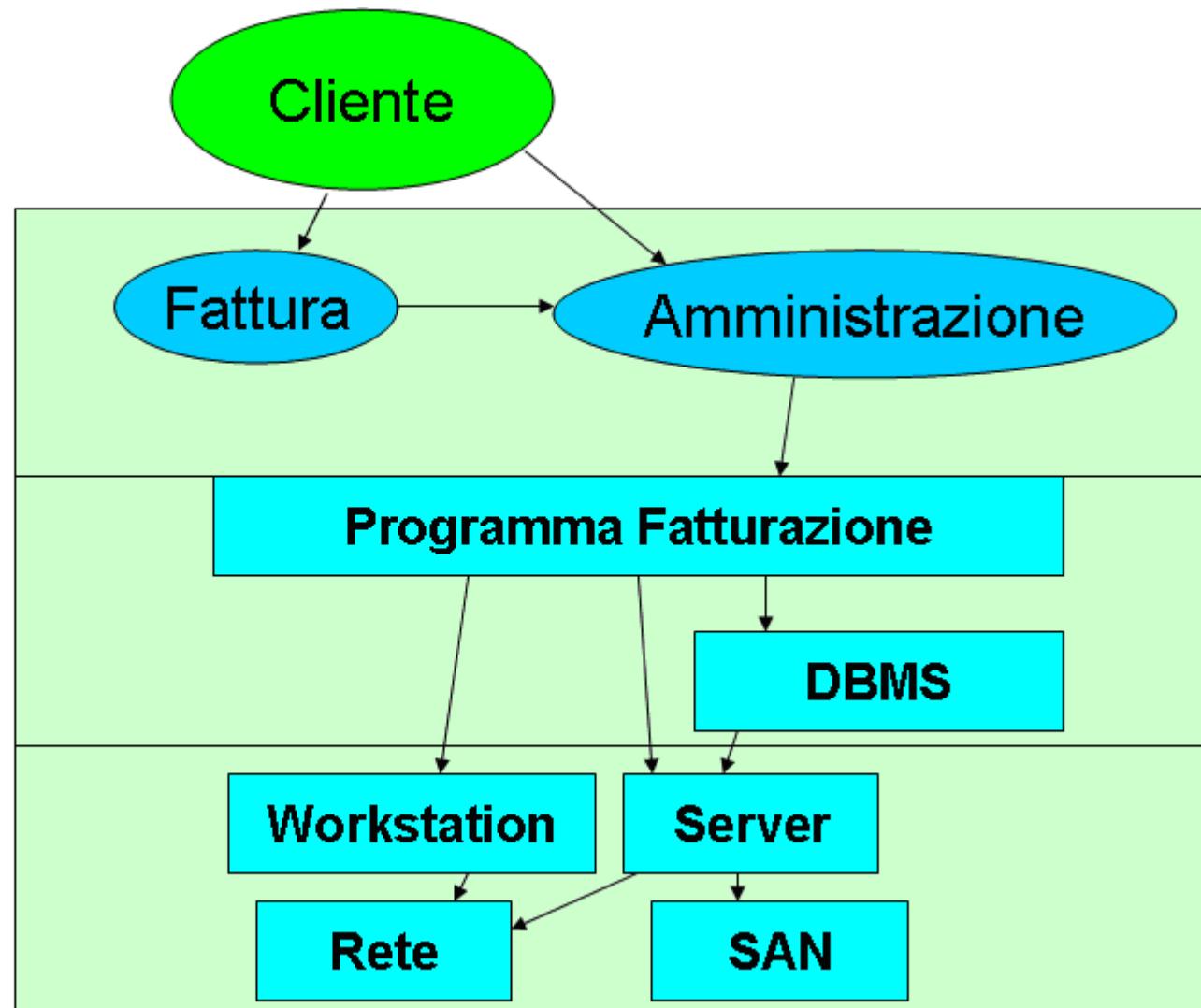
ITIL Service Catalogue: applicazione

Ambiente
Esterno

Livello
Business

Livello
Applicazioni

Livello
Tecnologia



Lo storage moderno e le architetture

*Affidabilità! Affidabilità! Affidabilità! ...
Ridondanza !*

Storage moderno: i dischi RAID

- Redundant Array of Inexpensive Disks
- Striping
- Diversi livelli

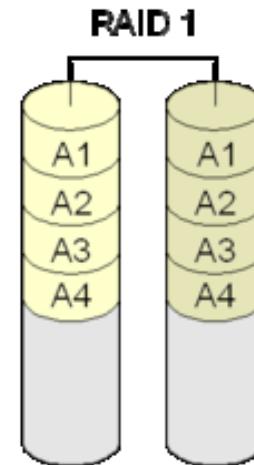
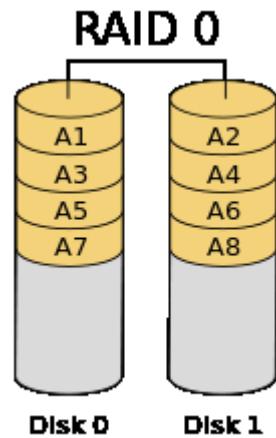
I dischi RAID: categorie (1/2)

- RAID 0: suddivisione (striping) senza ridondanza
- RAID 1: mirroring dei dischi
- RAID 2: striping + data check (su disco dedicato) a livello di bit (obsoleto)
- RAID 3: striping + data check (su disco dedicato) a livello di byte
- RAID 4: striping + data check (su 1 disco dedicato) a livello di blocco
- RAID 5: striping + parity check distribuito

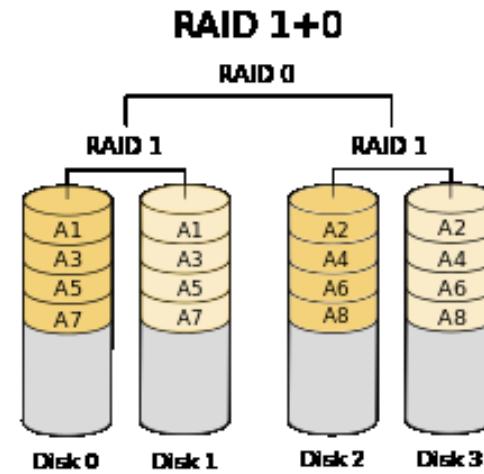
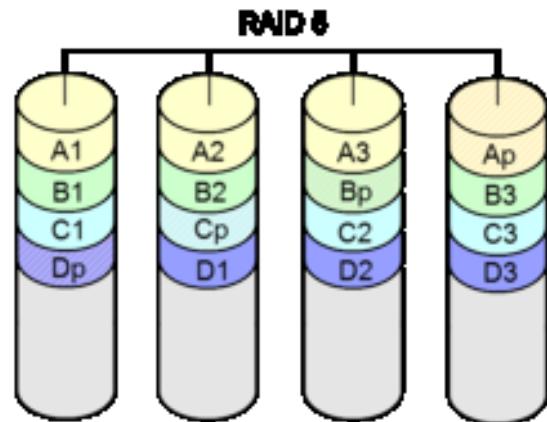
I dischi RAID: categorie (2/2)

- RAID 6: suddivisione dati fra dischi e parity check distribuito e duplicato
- RAID 0+1: sistema a due livelli, in cui un primo livello di RAID-0 viene duplicato ed organizzato in un RAID-1
- RAID 1+0 (detto anche RAID 10): sistema a due livelli, in cui vari sistemi RAID-1 vengono uniti in un RAID-0.
- A livello pratico sono normalmente in uso sistemi RAID 1, 2, 5 e 10.

Tipi di RAID più utilizzati



Hot-Swap
Hot-Spare

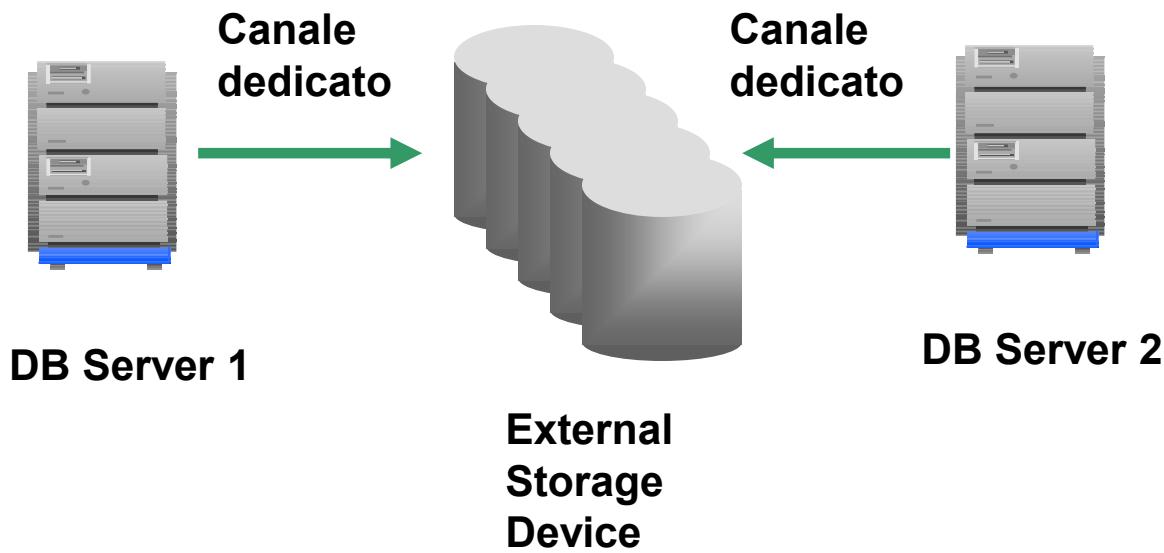


Tipi di strumenti per lo storage

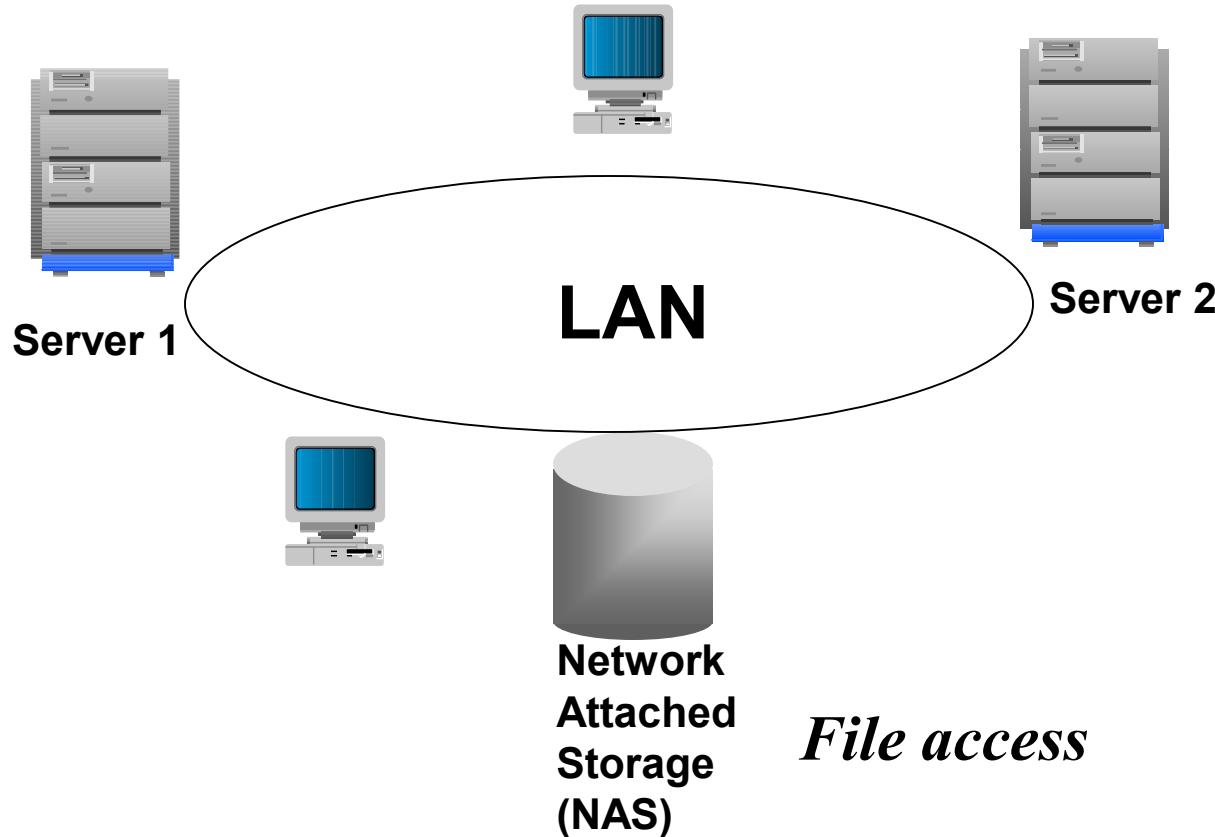
- Interni al server
- Storage device con “bus” dedicato (DAS)
- Network Attached Storage (NAS)
- Storage Area Network (SAN)

I dischi RAID – Storage Device

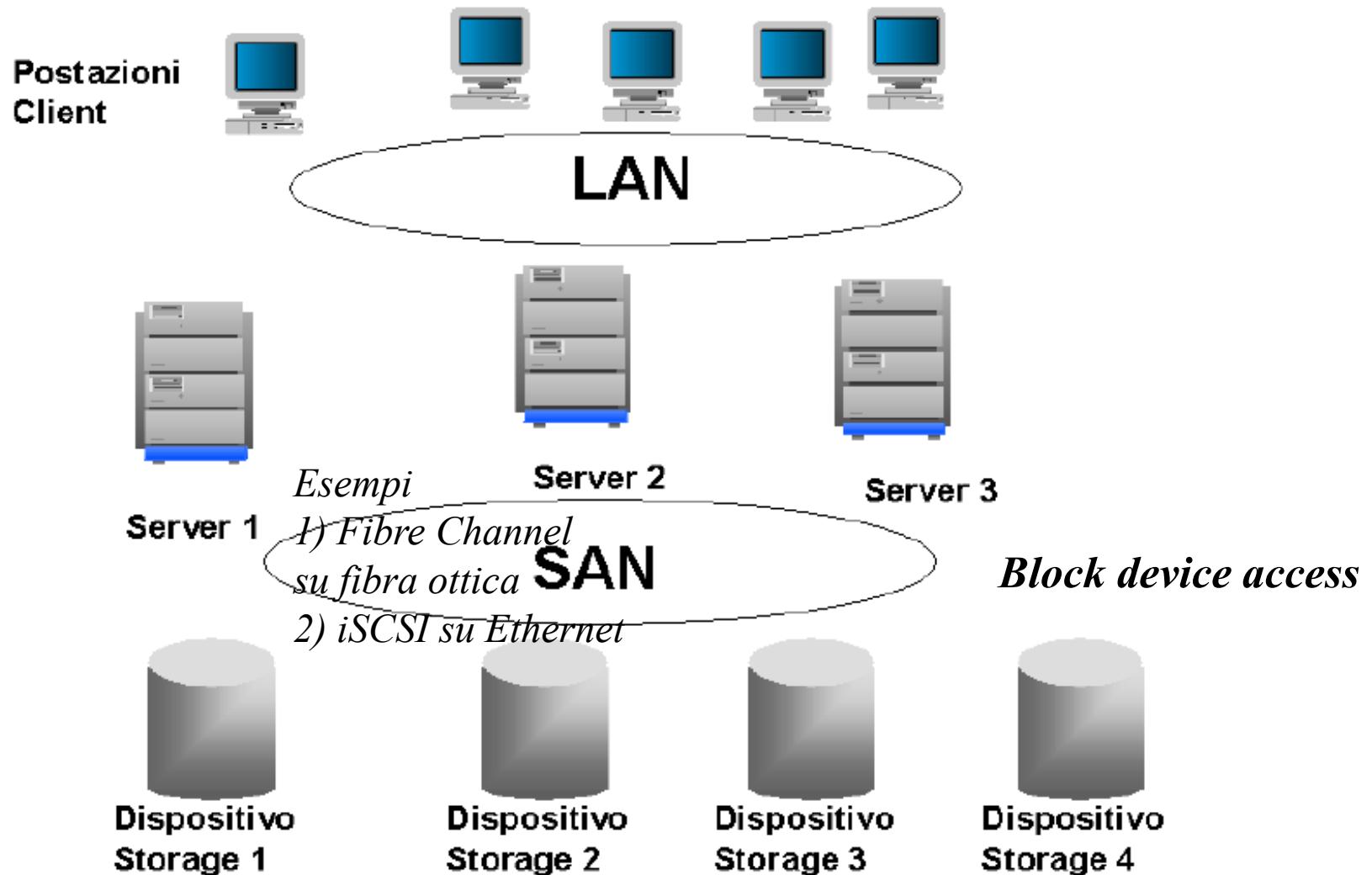
Direct Attached Storage (DAS)



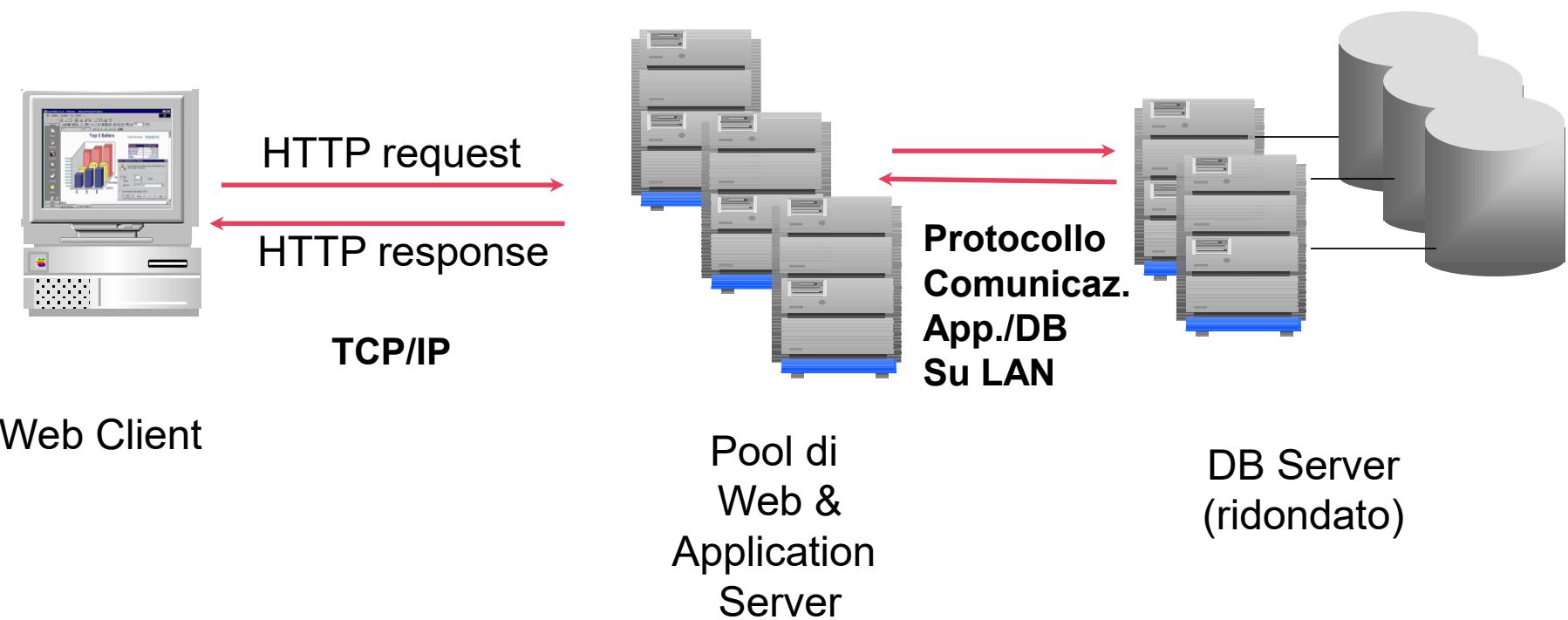
Network Attached Storage (Device)



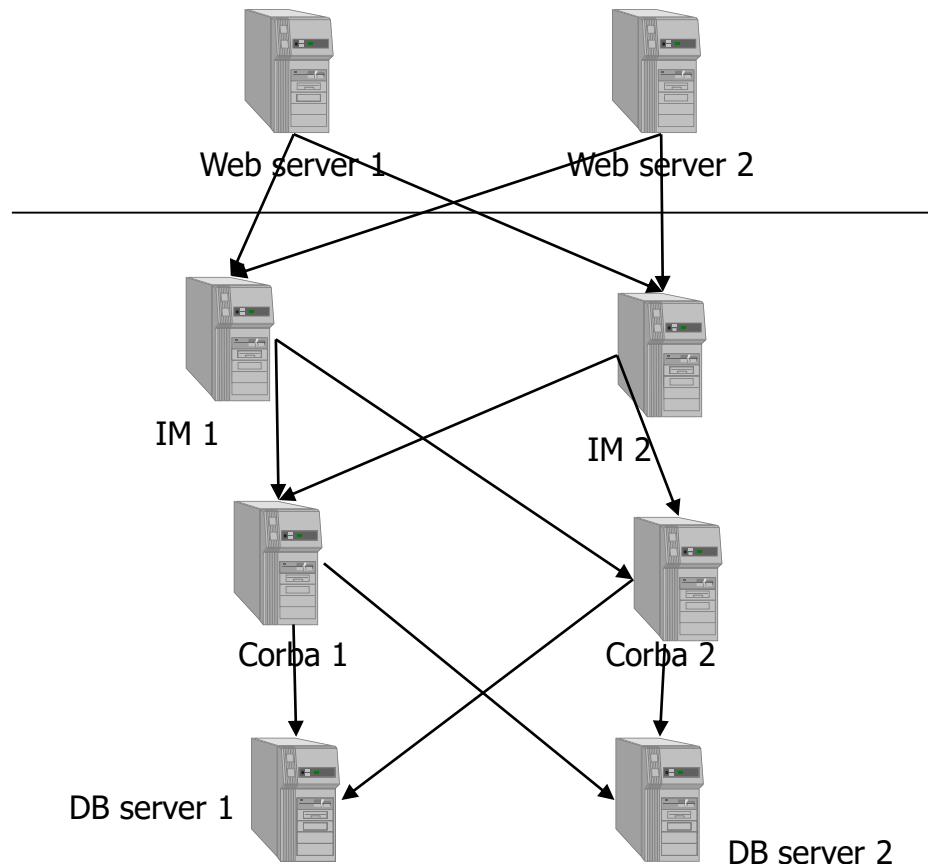
Storage Area Network (SAN)



Esempio di architettura ridondata



Duplicazione totale



Load Balancing

e

Fault tolerance

Sistemi Operativi e tecnologie per lo sviluppo di applicazioni

Sistemi Operativi Vari in uso in azienda

- (MS-DOS, DOS+Windows 3.x)
- Apple OSX
- Windows95/98/ME/XP/7/8/10
- WindowsNT/2000/Windows Server 20XY
- UNIX/Linux
- MVS, OS/390, Z/OS
- OS/400, I/OS
- VMS, OpenVMS

Linguaggi di sviluppo (1/2)

- Famiglia del COBOL
- FORTRAN
- La famiglia Pascal/Delphi
- Visual Basic
- Il mondo .NET
- Il mondo Java
- La famiglia C/C++

Linguaggi di sviluppo (2/2)

- I 4GL
- Il mondo assembler
- Il mondo Web
- RPG
- Administration scripting
- Altri... (Ruby, Python...)

Sistema informatico e applicazioni: integrazione mediante SOA

EAI: i tipi di integrazione

- Integrazione orientata ai dati
- Integrazione orientata a funzioni e metodi
- Integrazione di interfacce utente
- Integrazione dei processi business

EAI: integrazione orientata ai dati

- Avviene a livello di database o archivi dati
- Può essere real-time o no
 - Trasferimenti batch
 - Unioni di dati
 - Repliche di dati
 - Soluzioni ETL (Extract, Transform, Load)

EAI: integrazione di funzioni/metodi

- Integrazione di applicazioni (A2A)
 - Diretta, con paradigma request/response
-
- Basata su strumenti di middleware
 - O su codice custom

EAI: integrazione di interfacce (refacing)

- Standardizzazione delle interfacce utente entro un unico modello
- Di solito basata sul browser
- Enterprise business portal

EAI: integrazione dei processi business

- Agisce direttamente al livello dei processi business
- Non facile da applicare quando vi sono prodotti software con logica business rigida
- Per essere flessibile, conduce implicitamente alla SOA

Service-Oriented Architecture (SOA)

- La SOA è, ovviamente, un'architettura.
- E' più di un insieme particolare di tecnologie, come i Web Service, ed è definita indipendentemente da essi.
- Come il nome implica, i servizi sono il cuore della SOA

Ambiente di business: definizione di SOA

- Una architettura di applicazioni
 - entro la quale tutte le funzioni sono definite come servizi indipendenti
 - con interfacce invocabili ben definite,
 - che possono essere chiamate in sequenze definite
 - a formare i processi business
-
- I servizi sono il cuore della SOA

Ambiente di business: i servizi

- “Services are well-defined encapsulations of business assets” (Sun)
- I servizi sono accessibili via rete
- Descritti attraverso linguaggi standard per la definizione di interfacce
- Accessibili attraverso protocolli Web-based come SOAP

L'obiettivo della SOA

- La SOA si pone l'obiettivo di gestire
- la complessità,
- la mancanza di flessibilità,
- le problematiche di granularità
- legate agli approcci esistenti all'integrazione fra ambienti eterogenei e non.

Predecessori della SOA

- Tale obiettivo non è certamente nuovo
- Predecessori sono stati ad esempio
 - CORBA
 - MS DCOM
 - Ambienti proprietari (Tibco, BEA...)
- Ma ognuno con i propri limiti...

La struttura della SOA

- Topologia di applicazioni software
- formata da servizi e clienti dei servizi (service consumer)
- in relazione 1-a-1 tra di loro
- ma “debolmente accoppiati” (loosely coupled)



Componenti della SOA

- Service Provider (fornitore di un servizio)
- Service Requestor (cliente richiedente un servizio)
- Service Broker (intermediario)

SOA: service provider

Componente responsabile di

- Creare il servizio
- Pubblicare l'interfaccia del servizio
- Provvedere l'implementazione effettiva che realizza il servizio
- Rispondere alle richieste in arrivo (realizzare effettivamente il servizio)

SOA: service requestor (client)

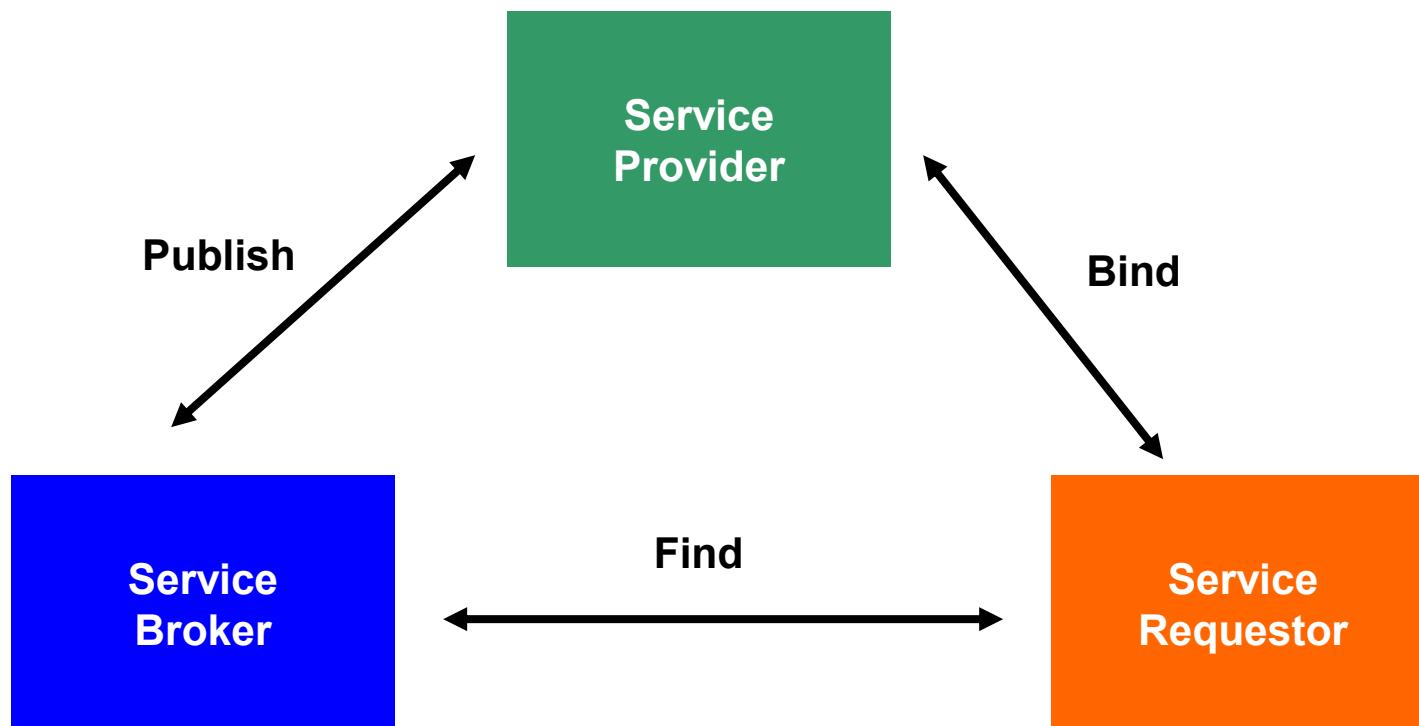
- Componente utente del servizio
- Deve trovare il servizio
 - Per conoscenza diretta
 - O interrogando un repository
- Deve inviare i dati previsti dall'interfaccia del servizio
- E ottenere indietro i risultati

SOA: service broker

- Componente intermediario
- Registra e categorizza i servizi
- Sono possibili interrogazioni con varie chiavi
- Crea e gestisce un repository di servizi

Operazioni nella SOA

- Pubblicazione (publishing) di servizi
- Reperimento (finding) di servizi
- Interfacciamento (binding) ai servizi



SOA e servizi effettivi

I servizi encapsulano tutta la gestione dello stato.

I servizi vengono chiamati tramite messaggi inviati con protocolli inaffidabili

Le sfide:

- Mantenimento dello stato della conversazione
- Gestione delle transazioni
- Gestione dello stato
- Data Caching per aumentare le performance, la scalabilità, e la disponibilità della soluzione

Funzionamento operativo della SOA

- I messaggi sono documenti di business scambiati per eseguire un processo di business.
- I servizi si aspettano che gli altri si ricordino delle conversazioni in atto.
- Esempio: identificativo acquirente

Funzionamento operativo della SOA

- Una conversazione basata su messaggi richiede che la situazione sia
 - salvata
 - e recuperata
 - quando necessario al processo di business
-
- ...come una conversazione tra persone

Tecnologie per SOA

- **Web Service basati su SOAP**
 - basato su XML+RPC su canale di trasporto (HTTP/HTTPS/SMTP/JMS/...)
 - WSDL per la descrizione delle funzionalità di un servizio
 - Numerosi standard WS-*, ad esempio WS-Security o WS-Addressing
- **Web Service RESTful** (Representational State Transfer)
 - uno stile architetturale, non uno standard
 - comunicano in HTTP/HTTPS con i verbi standard (GET, POST, PUT, DELETE, etc.)
 - le risorse sono identificate nelle richiesta mediante URI, e la loro rappresentazione restituita al client è codificata in JSON o XML

Esempio WSDL

```
<definitions name="HelloService"
targetNamespace="http://www.examples.com/wsdl/Hell
oService.wsdl"
    xmlns="http://schemas.xmlsoap.org/wsdl/"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:tns="http://www.examples.com/wsdl/HelloServic
e.wsdl"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">

    <message name="SayHelloRequest">
        <part name="firstName" type="xsd:string"/>
    </message>

    <message name="SayHelloResponse">
        <part name="greeting" type="xsd:string"/>
    </message>

    <portType name="Hello_PortType">
        <operation name="sayHello">
            <input message="tns:SayHelloRequest"/>
            <output
message="tns:SayHelloResponse"/>
        </operation>
    </portType>
```

```
<binding name="Hello_Binding" type="tns:Hello_PortType">
    <soap:binding style="rpc"
        transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="sayHello">
        <soap:operation soapAction="sayHello"/>
        <input>
            <soap:body
                encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
                namespace="urn:examples:helloservice"
                use="encoded"/>
        </input>

        <output>
            <soap:body
                encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
                namespace="urn:examples:helloservice"
                use="encoded"/>
        </output>
    </operation>
</binding>

<service name="Hello_Service">
    <documentation>WSDL File for
HelloService</documentation>
    <port binding="tns:Hello_Binding"
name="Hello_Port">
        <soap:address
            location="http://www.examples.com/SayHello/">
    />
    </port>
</service>
</definitions>
```

Esempio RESTful

Fake Online REST API for Testing and Prototyping (powered by JSON Server and lowdb) <http://jsonplaceholder.typicode.com/>

Showing a resource

```
$.ajax('http://jsonplaceholder.typicode.com/posts/1', {
  method: 'GET'
}).then(function(data) {
  console.log(data);
});
```

Listing resources

```
$.ajax('http://jsonplaceholder.typicode.com/posts', {
  method: 'GET'
}).then(function(data) {
  console.log(data);
});
```

Creating a resource

```
// POST adds a random id to the object
sent
$.ajax('http://jsonplaceholder.typicode.com/posts', {
  method: 'POST',
  data: {
    title: 'foo',
    body: 'bar',
    userId: 1
  }
}).then(function(data) {
  console.log(data);
});

/* will return
{
  id: 101,
  title: 'foo',
  body: 'bar',
  userId: 1
}
*/
```

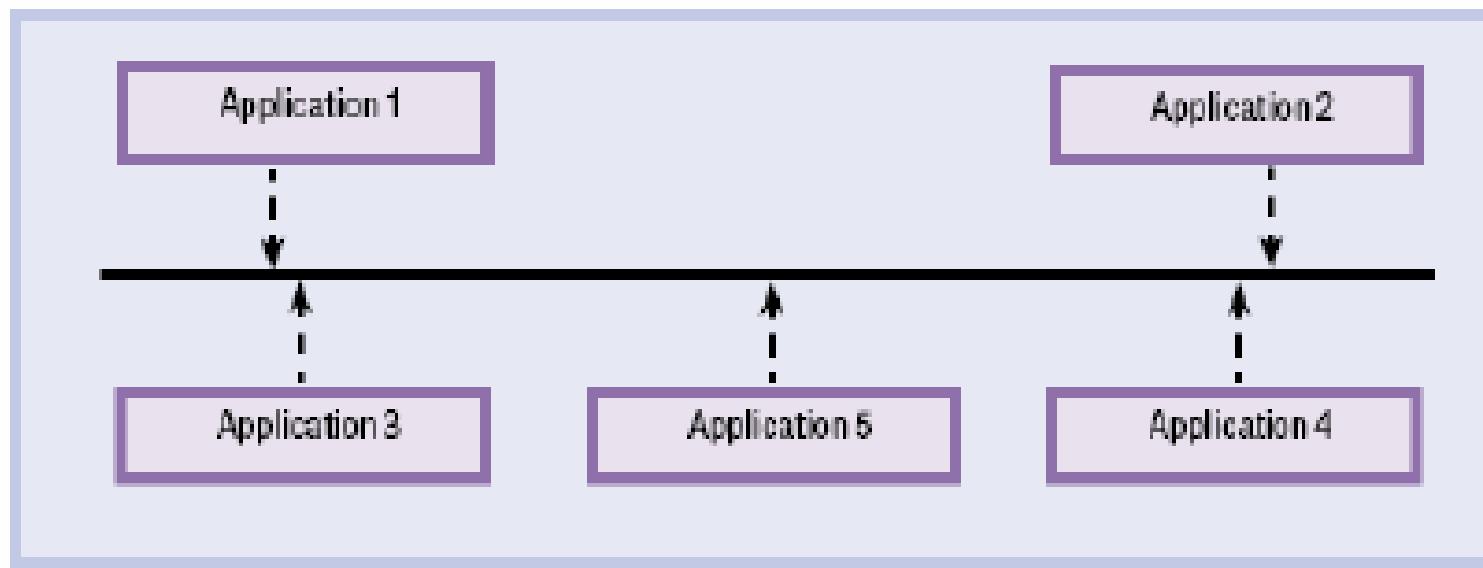
Updating a resource

```
$.ajax('http://jsonplaceholder.typicode.com/posts/1', {
  method: 'PUT',
  data: {
    id: 1,
    title: 'foo',
    body: 'bar',
    userId: 1
  }
}).then(function(data) {
  console.log(data);
});

/* will return
{
  id: 1
  title: 'foo',
  body: 'bar',
  userId: 1
}
```

L'integrazione dei servizi nella SOA tradizionale: l'Enterprise Service Bus

- Per garantire il disaccoppiamento e minimizzare il numero di interfacce
- Il componente intermediario deve assumere un ruolo “attivo”



L' Enterprise Service Bus (ESB)

- L'ESB è una soluzione di integrazione basata su standard aperti, message-based, distribuita
- che provvede a routing, invocazione e mediazione
- tra i servizi per facilitare le interazioni fra risorse IT distribuite (applicazioni, servizi, informazioni, piattaforme)
- in modo affidabile.
(Brenda M. Michealson)

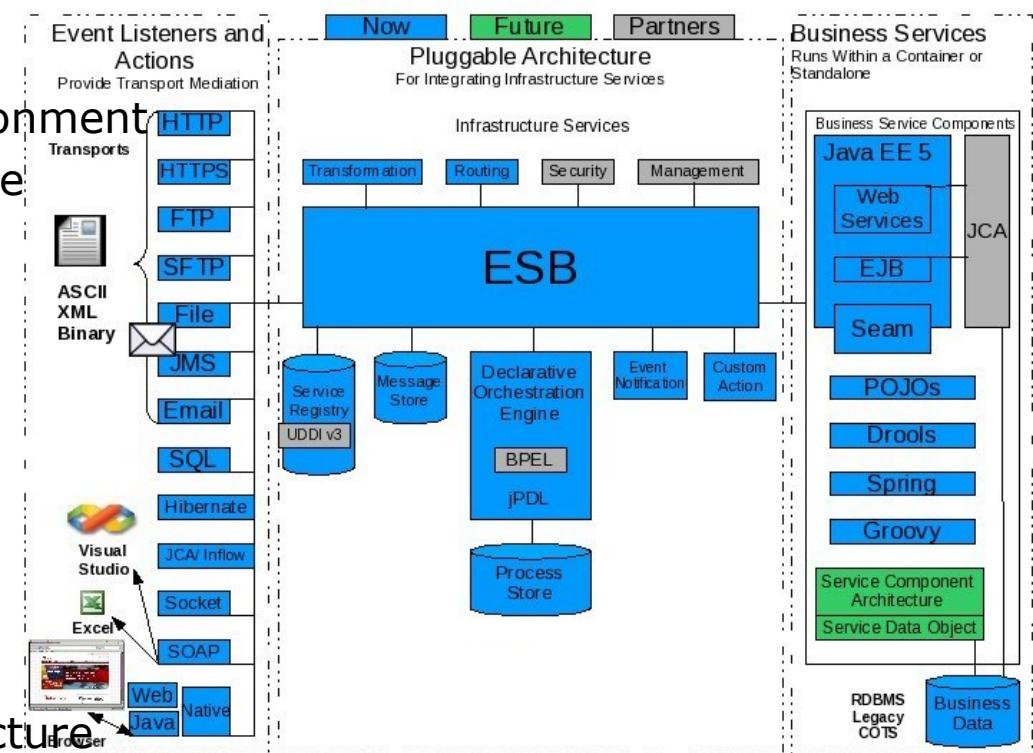
Punti chiave per l'ESB

- **Standard aperti**
- **Message-Based**: messaging, usando notazioni di messaggio, protocolli e trasporti standard
- **Distribuito**: Il runtime environment dell' ESB può essere distribuito (QoS, scalabilità...)
- **Routing, invocazione e mediazione**
- **Facilitare**
- **Affidabilità**

JBoss ESB (RedHat)

Soluzione open-source di ESB per EAI che include:

- Business Process Monitoring
- Integrated Development Environment
- Human Workflow User Interface
- Business Process Management
- Connectors
- Transaction Manager
- Security
- Application Container
- Messaging Service
- Metadata Repository
- Naming and Directory Service
- Distributed Computing Architecture



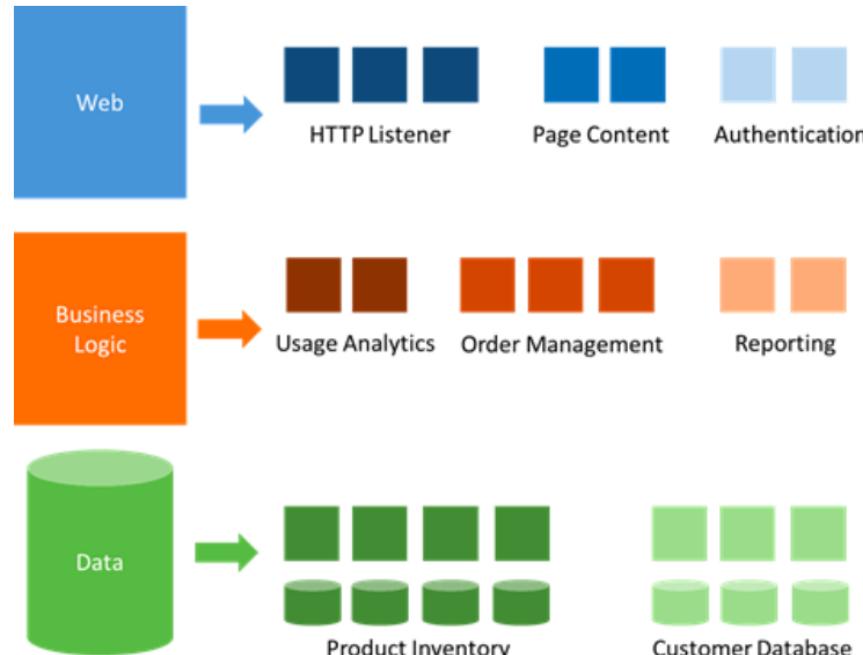
*Il supporto di RedHat non durerà a lungo: un ESB più recente è **JBoss Fuse** che è orientato ai microservizi*

SOA moderna: microservices

Una variante delle SOA che struttura le applicazioni in collezioni di servizi:

- a grana fine
- protocolli standard, sincroni (es. HTTP/REST) o asincroni (es. AMPQ,RabbitMQ)

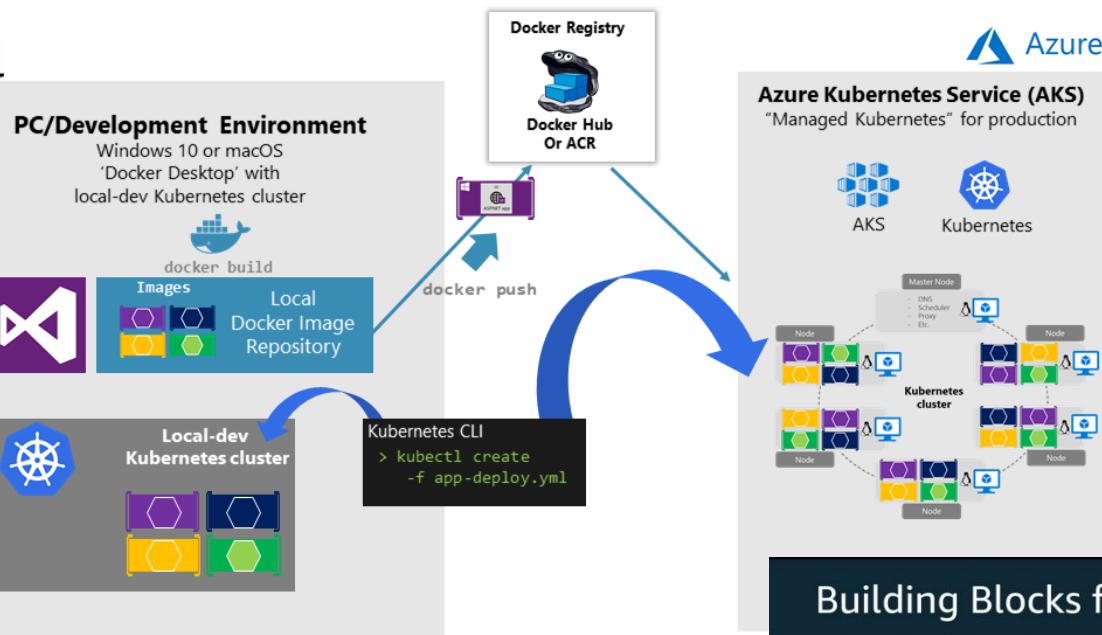
Le applicazioni monolitiche sono complesse e più problematiche da sviluppare, manutenere e sostituire nell'epoca del 24x7



I servizi possono essere sviluppati singolarmente e individualmente deployable e sostituiti

Più scalabili grazie ai container e al cloud

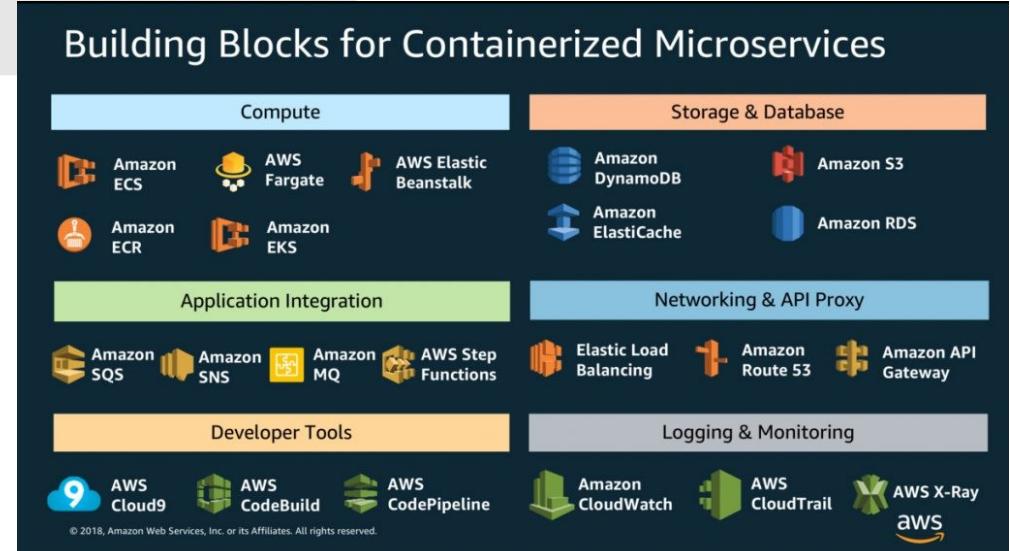
Microservizi, container e Kubernetes



Microsoft Azure

Amazon AWS

Amazon Elastic Kubernetes Service (Amazon EKS) è un servizio Kubernetes interamente gestito



Dai Cluster al Cloud Computing

Just because we continue to see new innovations in IT does not mean that it pays to be a pioneer.

Nicholas G. Carr

Cluster

“It is quite possible that by the middle of this decade clusters in their myriad forms will be the dominant high-end computing architecture”

Dr. Thomas Sterling

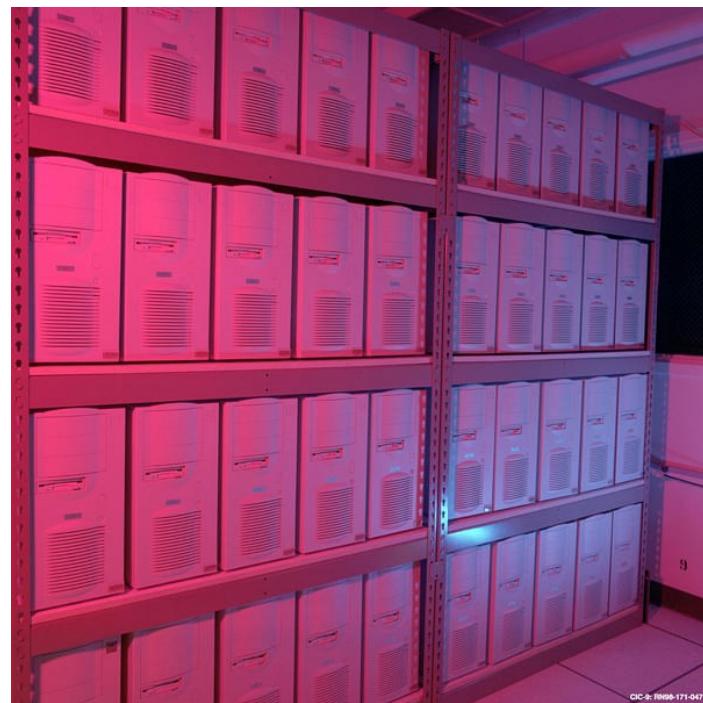
California Institute of Technology

and the

NASA Jet Propulsion Laboratory

Progettista dei primi sistemi CLUSTER “Beowulf” anno 1990 circa

Sistemi a Cluster (anni '90)



I moderni sistemi ad elevate prestazioni sfruttano la disponibilità di nodi di calcolo con alte prestazioni disponibili sul mercato a costi ridotti e utilizzano per i sistemi di interconnessione le metodologie derivate da INTERNET

Sistemi a Cluster : IBM SP Power 6

Modello: IBM pSeries 575

Architettura: IBM P6-575 Infiniband Cluster

Processore : IBM Power6, 4.7 GHz

Computing Cores: 5376

Nodi di calcolo: 168

RAM: 1.2 TB (128 GB/node)

Rete interna: Infiniband x4 DDR

Spazio disco: 21 PB

O.S.: AIX 6

Prestazioni di picco: 101 TFlop/s

Compilatori: Fortran90, C, C++

Librerie parallele: MPI, OpenMP, LAPI

CINECA



Nel 2005 al 46° posto nella classifica c

Nel 2012 sostituito da FERMI IBM-BG/Q , a sua volta sostituito nel 2016 da MARCONI –A1 (acceleratori Intel Xeon Phi)

(<http://www.hpc.cineca.it/content/hardware>)

HPC@CINECA

Cineca is currently one of the **Large Scale Facilities** in Europe and it is a [PRACE](#) Tier-0 hosting site.

- **MARCONI:** It is the **new Tier-0** system that replaced FERMI in July 2016. It is based on the LENOVO NeXtScale platform and the next generation of the Intel Xeon Phi product family. It has been gradually completed in about 12 months and now it is in its final configuration: the first partition (**Marconi-A1**) with Intel Broadwell, into production from July 2016, has been closed on Sept 2018; the second partition (**Marconi-A2**) with Intel KnightsLanding is into production from January 2017, the third partition (**Marconi-A3**) with SkyLake is into production from August 2017, upgraded on Jan 2018 and finally on Nov 2018. Marconi is classified in **Top500 list** among the most powerful supercomputer: **rank 12** in November 2016, and rank 19 in the November 2018 list.
- **GALILEO:** It has been renewed in March 2018 with **Intel Xeon E5-2697 v4 (Broadwell) nodes**, available for italian research community.
- **D.A.V.I.D.E.:** (**D**evelopment of an **A**dded **V**alue **I**nfrastructure **D**esigned in **E**urope) It is the energy-aware, High Performance Cluster, based on OpenPOWER8 servers and NVIDIA Tesla P100 data center GPUs. It entered the **Top500** and **Green500lists** in June 2017, and **entered into production in January 2018**.

HPC@CINECA

	CPU (mhz,core, ...)	Total cores / Total Nodes	Memory per node	Accelerator	Notes
MARCONI-A3	Intel SkyLake 2x Intel Xeon 8160 @2.1GHz 24 cores each	72576+38016+ 43776/ 1512+792+912	192 GB	-	
GALILEO	Intel Broadwell 2x Intel Xeon E5-2697 v4 @2.3GHz 18 cores each	12960 / 360	128 GB		
D.A.V.I.D.E.	OpenPOWER8 NVIDIA Tesla P100 SXM2 @2GHz 16 cores each	720 / 45		Tesla P100	

HPC@CINECA



MARCONI - 100

Nodes: 980

Processors: 2x16 cores IBM POWER9 AC922 at 3.1 GHz

Accelerators: 4 x NVIDIA Volta V100 GPUs, Nvlink 2.0, 16GB

Cores: 32 cores/node

RAM: 256 GB/node

Peak Performance: ~32 PFlop/s

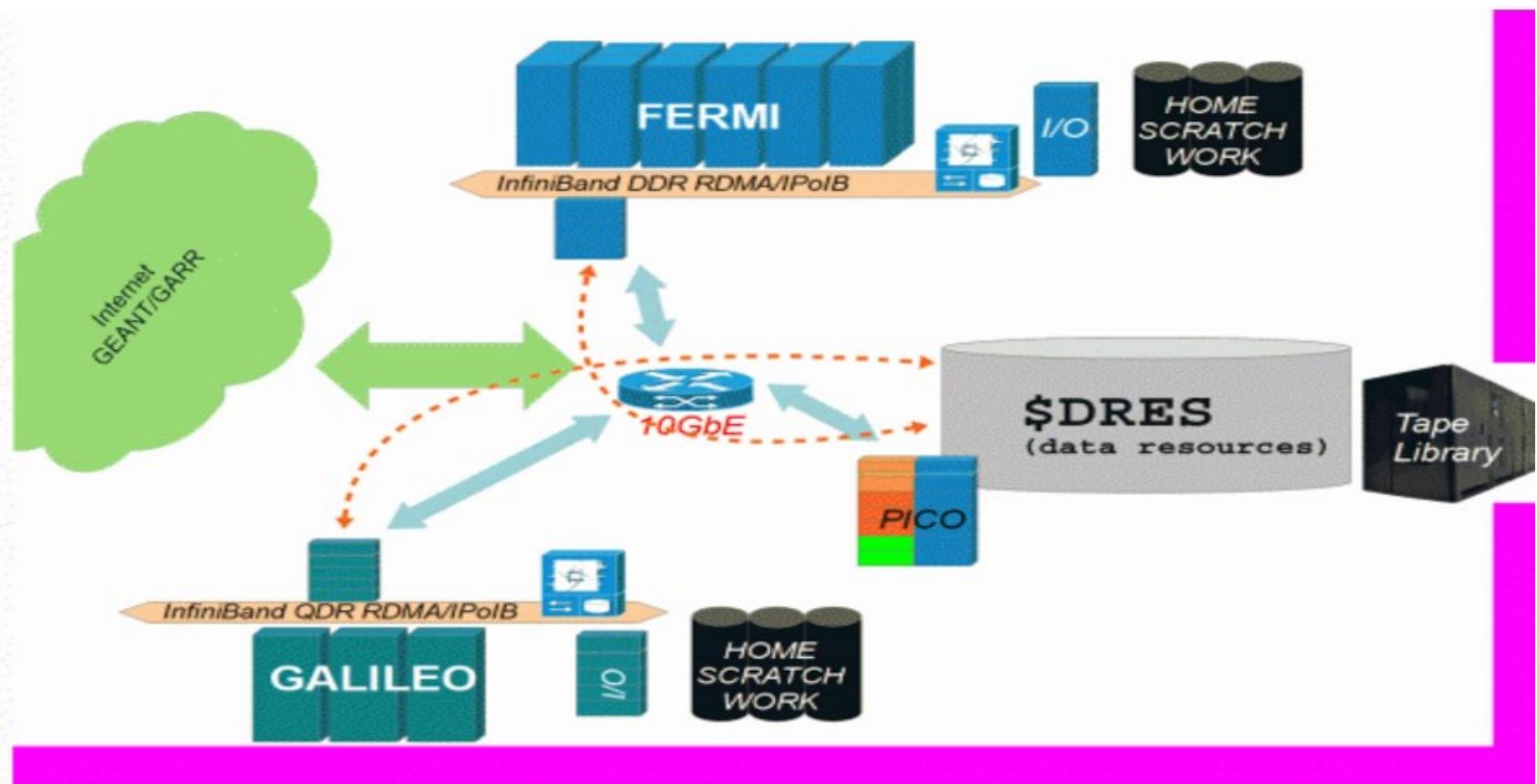
MARCONI 100 is the new accelerated cluster based on IBM Power9 architecture and Volta NVIDIA GPUs, acquired by Cineca within PPI4HPC European initiative. This system opens the way to the pre-exascale **Leonardo Supercomputer** (<https://www.cineca.it/temi-caldi/Leonardo>) expected to be installed in 2021 (peak performance 200+ PFlop/s, 3+ PB of RAM, 150 PB I/O, 150PB of storage.) It is available from April 2020 to the Italian public and industrial researchers. Its computing capacity is about 32 PFlops.

HPC@CINECA – Data storage

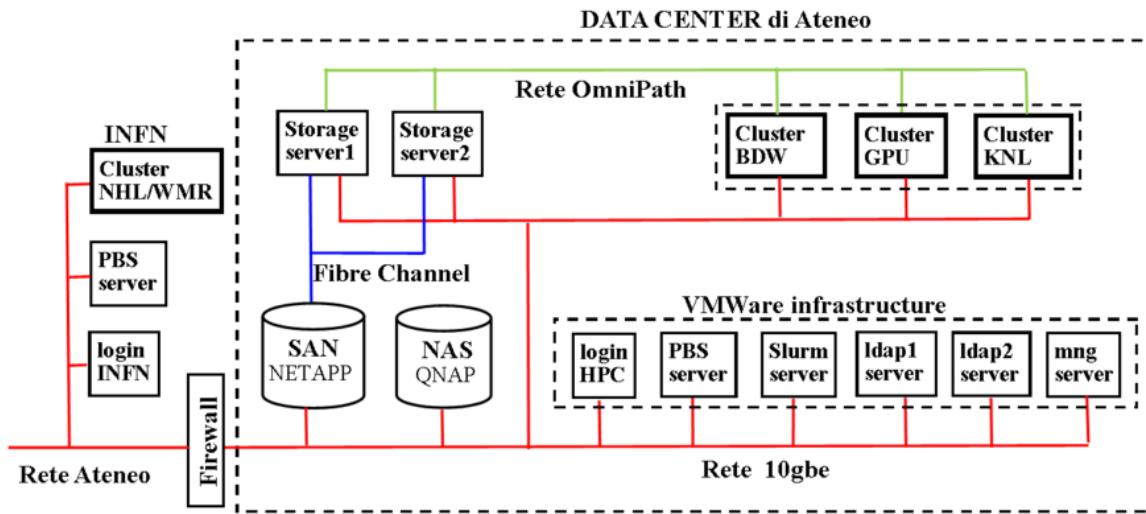
- **Scratch:** each system has its own local scratch area (pointed by \$CINECA_SCRATCH env variable)
- **Work:** a working storage is mounted to the three systems (pointed by \$WORK env variable)
- **DRes:** a shared storage area is mounted on all machine's login-nodes and all Pico's computes node (pointed by \$DRES env variable)
- **Tape:** a tape library (12 PB, expandible to 16PB) is connected to the DRES storage area as a multi-level archive (via [LTFS](#))

	Scratch (local)	Work (local)	DRes (shared)	Tape (shared)
<i>Fermi</i>	900 TB	1800 TB		
<i>MARCONI</i>	tbd (10PB in total)	tbd (10PB in total)	4 PB	12 PB
<i>GALILEO</i>	300 TB	1500 TB		
<i>Pico</i>	100 TB	500 TB		

HPC@CINECA – Data storage



Cluster HPC@UNIPR



Nuovi nodi calcolo

- Cluster1 ([BDW](#))
 - 8 nodi con 2 Intel Xeon E5-2683v4 (2×16 cores, 2.1GHz, 40MB smartcache), 128 GB RAM.
 - 1 nodo con 2 Intel Xeon E5-2683v4 (2×16 cores, 2.1GHz, 40MB smartcache), 1024 GB RAM.
- Cluster2 ([GPU](#))
 - 2 nodi con 2 Intel Xeon E5-2683v4 (2×16 cores, 2.1GHz), 128 GB RAM, 5 GPU NVIDIA P100-PCIE-12GB (Pascal architecture).
- Cluster3 ([KNL](#))
 - 4 nodi con 1 Intel Xeon PHI 7250 (1×68 cores, 1.4GHz, 16GB MCDRAM), 192 GB RAM.

Dettaglio dei nodi:Prestazioni di picco (doppia precisione):

1 Nodo BDW -> 2x16 (cores) x 2.1 (GHz) x 16 (AVX2) = 1 TFlops, Max memory Bandwidth = 76.8 GB/s

1 GPU P100 -> 4.7 TFlops

1 nodo KNL -> 68 (cores) x 1.4 (GHz) x 32 (AVX512) = 3 TFlops, Max memory bandwidth = 115.2 GB/s

Interconnessione con [Intel OmniPath](#) Prestazioni di picco: Bandwidth: 100 Gb/s, Latency: 100 ns.

Cluster

CLUSTER:

classe di architetture di elaborazione parallele che si basa su un insieme di elaboratori indipendenti cooperante per mezzo di una rete di interconnessione e coordinato mediante un sistema operativo che lo rende in grado di operare su di un singolo workload.

COMMODITY CLUSTER:

è costituito da nodi di elaborazione commerciali (COTS – Commercial off the shelf) in grado di operare in modo indipendente collegati tramite una rete di interconnessione (LAN o SAN) anch'essa di tipo COTS.

Classi di applicazioni

Un cluster può essere utilizzato in molti modi (ad esempio):

- elevate prestazioni su di un singolo problema,
- elevato throughput o elevata capacità con un carico di lavoro a processi,
- elevata disponibilità mediante la ridondanza dei nodi,
- elevata banda ottenuta dalla molteplicità dei dischi o dei canali di I/O.

Cluster

Che cosa giustifica l'interesse così forte per i sistemi a cluster:

Buon rapporto prestazioni/prezzo:

i commodity cluster possono migliorare il rapporto prestazioni/prezzo di un ordine di grandezza rispetto da architetture diverse.

Rafforzamento del ruolo del personale interno:

Sistemi cluster ad elevate prestazioni utilizzano moduli elementari che derivano dai sistemi di uso personale.

L'amministrazione, la manutenzione HW e SW richiedono competenze maggiormente disponibili sul mercato.

Si ha un maggior senso di controllo del sistema e della sua evoluzione.

Cluster

Tempi e costi di sviluppo:

I cluster si basano su elementi (nodi di elaborazione e sistemi di rete) disponibili sul mercato spesso per uso personale. L'integrazione non richiede moduli specializzati .

Convergenza dell'architettura:

Per la prima volta utenti e venditori hanno a disposizione un'architettura parallela stabile e indipendente.

Questo garantisce persistenza a lungo termine dell'architettura e giustifica investimenti software.

Architettura

Tracking della tecnologia:

I'uso di moduli standard consente di inserire nell'architettura gli ultimi prodotti della tecnologia. Questo può avvenire assai prima che in altre architetture parallele e può avvenire anche parzialmente.

Scalabilità:

le «nuove» infrastrutture di rete locale utilizzate (switched Ethernet, Fast o Gigabit Ethernet) consentono di collegare tra loro un numero molto elevato di nodi senza inserire colli di bottiglia hardware significativi e mantenendo la stessa impostazione architetturale.

Architettura

Flessibilità della configurazione e degli aggiornamenti:

le possibili architetture in termini di topologia, distribuzione fisica, localizzazione geografica sono estremamente varie e anche gli aggiornamenti possono essere mirati e specifici (dove servono e quando servono).

Alta disponibilità (availability):

- L'Alta Disponibilità è ortogonale alle Alte Prestazioni
- Alcuni (tutti ?) servizi devono essere disponibili 24 x 365
- Devono essere identificati i Single Point of Failure (ad es. la scheda di rete).

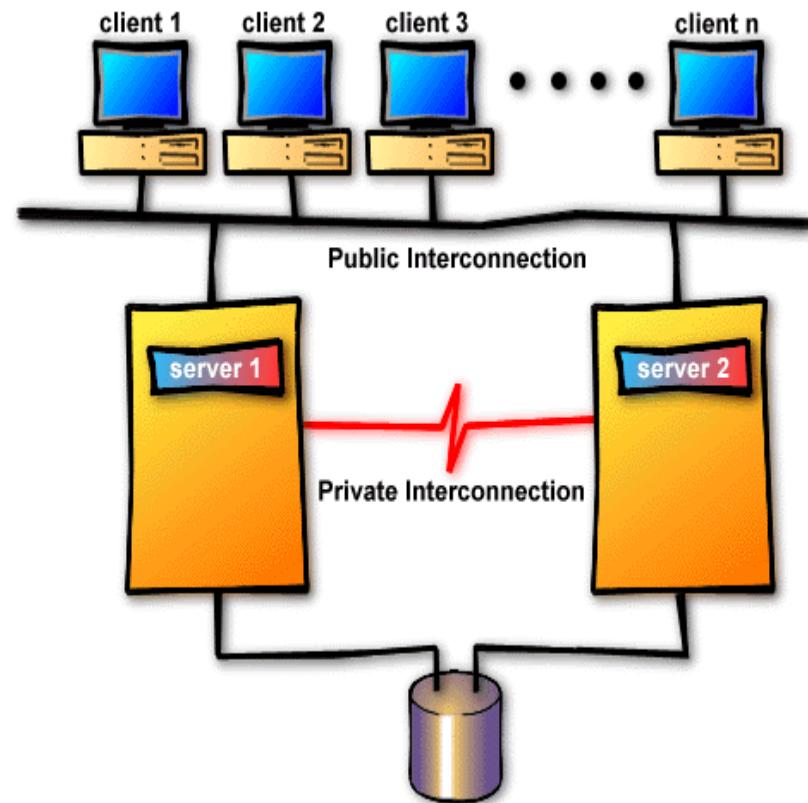
Esempio: High Availability (HA)

I clienti raggiungono il servizio a uno specifico indirizzo IP (all'inizio associato al server primario).

Con due schede di rete per nodo è possibile trasferire dinamicamente l'indirizzo del servizio :

- da una scheda all'altra (fallimento della scheda)
- da un server all'altro (fallimento del nodo)

L'operatività dei nodi è verificata mediante un dialogo continuo su un canale dedicato (*heartbeat*)

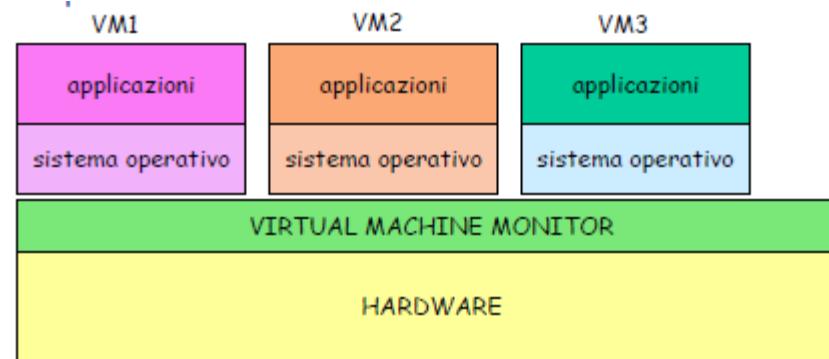


Virtualizzazione

- Con **virtual machine** si intende un ambiente software emulato su una macchina.
- La Virtual Machine emula un ambiente in cui sono installati altri programmi, si hanno così istanze di sistemi operativi differenti: su una macchina Window sarà possibile avviare una VM dedicata all'esecuzione di Linux e di un particolare servizio, oppure potranno essere avviate varie istanze per simulare complesse strutture server.
- Utilizzando una Virtual Machine non si ha l'esigenza di disporre di svariate macchine fisiche per i test, commutare da una configurazione all'altra sarà praticamente un'azione istantanea.
- Un limite all'uso delle virtual machine è però dato dalle risorse hardware necessarie a garantire un funzionamento efficace. I pc di recente fabbricazione di fascia media riescono a garantire i requisiti hardware necessari all'emulazione di un ambiente.

Virtualizzazione di Sistema

- Una singola piattaforma hardware viene condivisa da più sistemi operativi, ognuno dei quali è installato su una diversa macchina virtuale.



- Il disaccoppiamento è realizzato da un componente chiamato Virtual Machine Monitor (VMM, o hypervisor) il cui compito è consentire la condivisione da parte di più macchine virtuali di una singola piattaforma hardware. Ogni macchina virtuale è costituita oltre che dall'applicazione che in essa viene eseguita, anche dal sistema operativo utilizzato.
- Il VMM è il mediatore unico nelle interazioni tra le macchine virtuali e l'hardware sottostante, che garantisce:
 - **isolamento** tra le VM
 - **stabilità** del sistema

VMM di sistema vs. VMM ospitati

VMM di Sistema

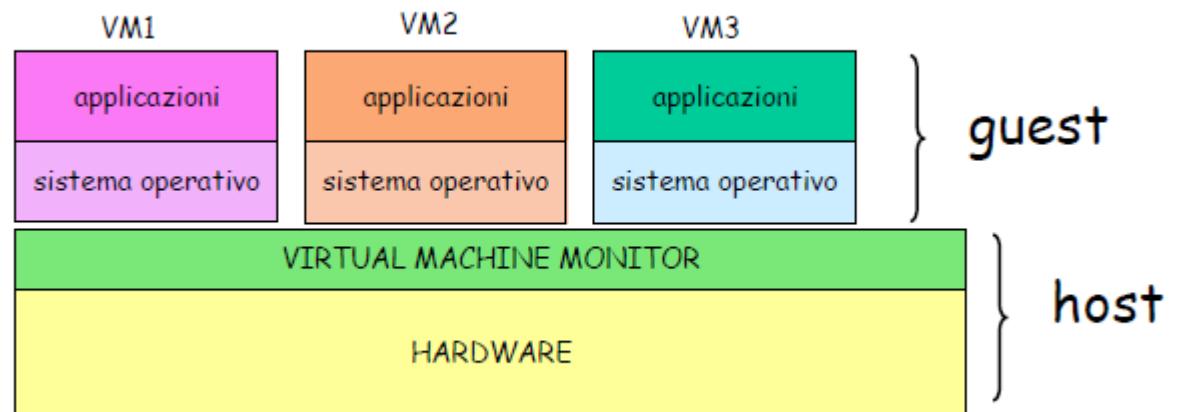
- le funzionalità di virtualizzazione vengono integrate in un sistema operativo leggero, costituendo un unico sistema posto direttamente sopra l'hardware dell'elaboratore.
- E' necessario corredare il VMM di tutti i driver necessari per pilotare le periferiche.
- Esempi di VMM di sistema: VMware ESX, Xen, VirtualIron.

VMM di Sistema

- **Host:** piattaforma di base sulla quale si realizzano macchine virtuali.

Comprende la macchina fisica, l'eventuale sistema operativo ed il VMM

- **Guest:** sistema



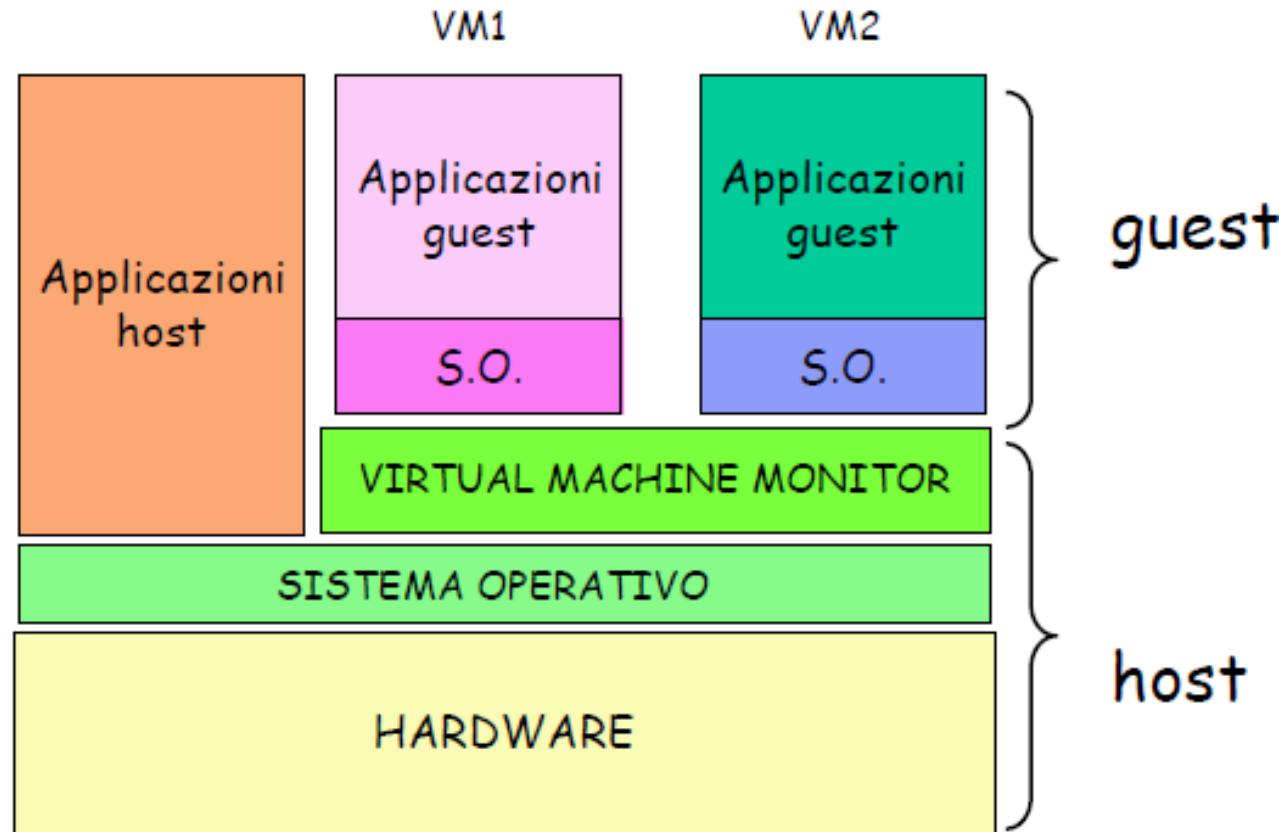
VMM di Sistema

VMM ospitato

Il VMM viene installato come un'applicazione sopra un sistema operativo esistente, che opera nello spazio utente e accede l'hardware tramite le system call del S.O. su cui viene installato.

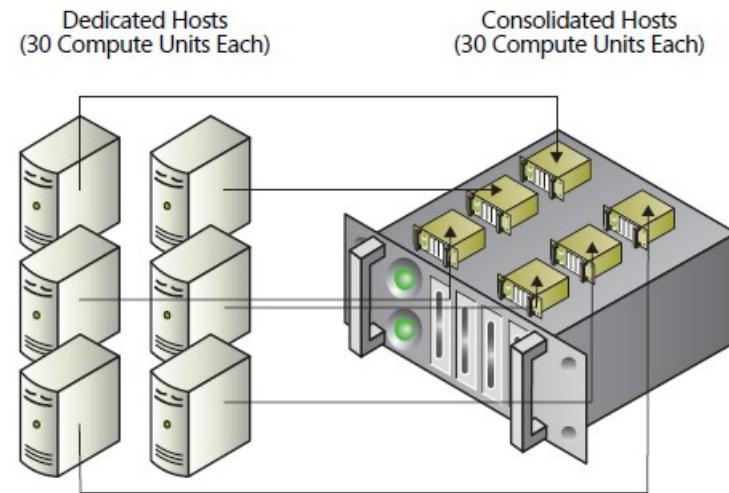
- Più semplice l'installazione (come un'applicazione).
- Può fare riferimento al S.O. sottostante per la gestione delle periferiche e può utilizzare altri servizi del S.O.(es. scheduling, gestione delle risorse.).
- Peggiora la performance.
- Prodotti: User Mode Linux, VMware Server/Player, Microsoft Virtual Server, Parallels

VMM ospitato



Virtualizzazione

- Il concetto di **macchina virtuale** fino a qualche tempo fa apparteneva al mondo degli sviluppatori o degli ambienti server e più anticamente dei vecchi mainframe.
- Si parla di **consolidamento** della struttura quando si centralizzano vari servizi su un unico server: i differenti servizi vengono eseguiti da varie virtual machine eseguite dalla stessa macchina fisica.



Questa soluzione permette un razionale uso dell'hardware, ottimizzando la gestione delle risorse in funzione delle esigenze istantanee. Prevedere una struttura ridondante diventa più semplice poiché a livello fisico si tratta di replicare una singola macchina, su cui andranno clonate le istanze virtuali.

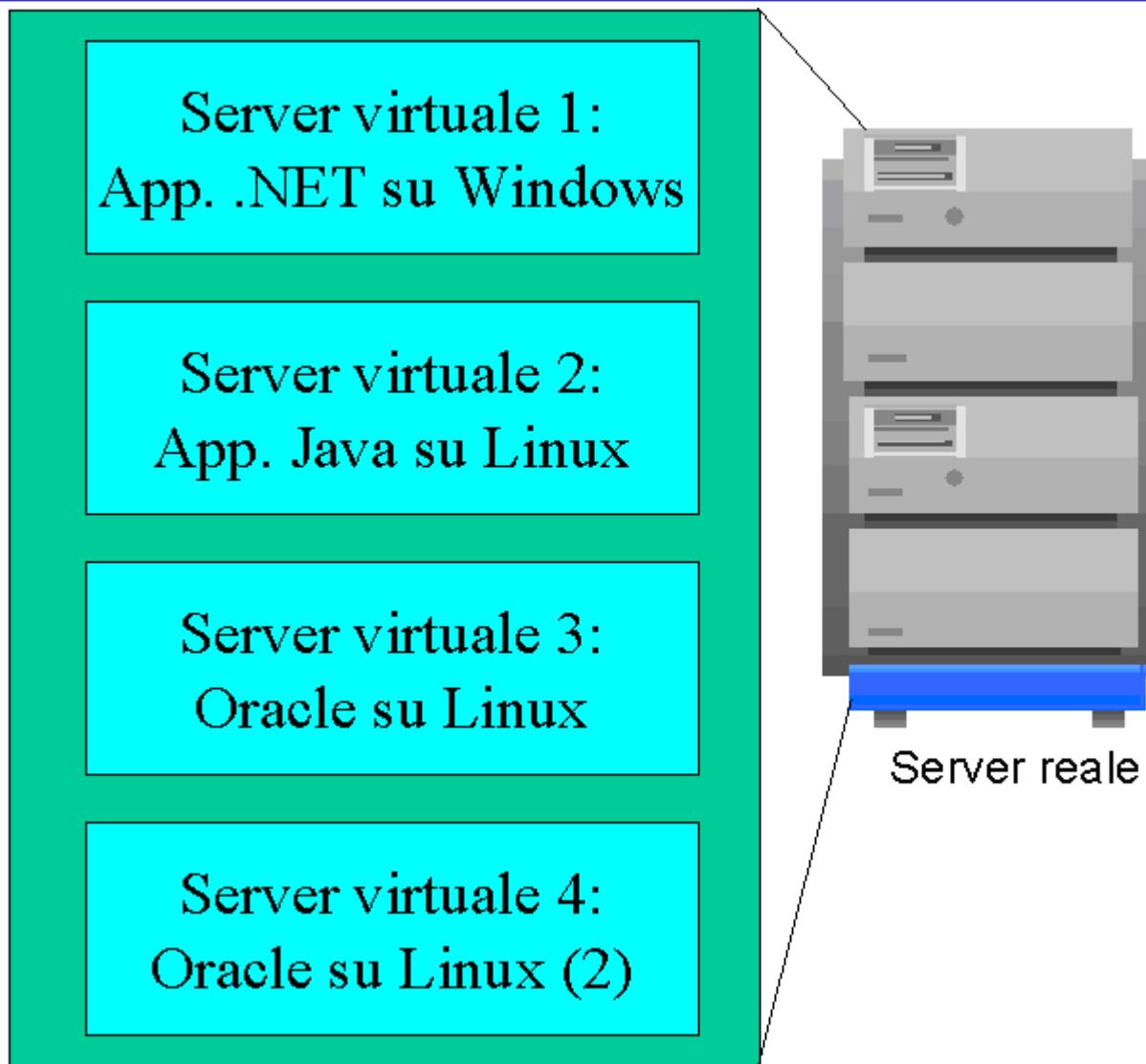
Virtualizzazione

- La virtualizzazione riduce il tempo per attivare nuovi servizi. L'avvio di una nuova VM richiede 30 minuti, mentre l'installazione di un nuovo server richiede settimane. I server fisici da gestire sono di meno, ma il tempo da dedicare alla gestione cresce: l'infrastruttura virtuale va gestita e può richiedere anche un nuovo sistema di storage centralizzato.
Problema: calcolo dei costi delle licenze software.
- Il primo obiettivo della virtualizzazione è il consolidamento. Su 10 server si svolge lo stesso lavoro che prima richiedeva 100 server.
“ogni nuovo carico di lavoro deve andare su una virtual machine”, ma per spostare un vecchio processo da una macchina fisica a una virtuale occorre conciliare piani e scadenze degli utenti, e saper affrontare la possibilità dell'indisponibilità delle applicazioni.

Virtual Server

- Computer server operante entro una virtual machine
- Può essere spostato da un server fisico ad un altro, spesso anche “a caldo”
- Il ripristino delle sue funzionalità quasi sempre è il semplice ripristino dall’immagine (snapshot)

Server virtuali su un server fisico



Desktop Virtualization

- Tecnologia che separa l'ambiente desktop di una postazione di lavoro con le sue applicazioni dal client fisico usato per accederlo
- Le postazioni fisiche mostrano in desktop remoto i desktop degli ambienti di lavoro (entro macchine virtuali o sessioni su server)

Virtualizzazione

- La virtualizzazione riduce automaticamente i consumi di energia?
E' vero che i server sono in numero minore, ma ciascuno ha un tasso più alto di utilizzo della CPU, e richiede più energia.
Il centro di calcolo deve essere riconfigurato per evitare di sprecare aria condizionata su spazi vuoti.
Inoltre è più facile attivare nuovi servizi e quindi ...
- Virtualizzazione e sicurezza
E' possibile 'clonare' le VM e trasferirle da una macchina all'altra facilitando le strategie di *disaster recovery*, e riducendo i rischi di perdite di dati e indisponibilità dei servizi.
La proliferazione dei server virtuali richiede che tutti siano aggiornati e questo è sicuramente impegnativo.

Virtualizzazione

- La virtualizzazione non è solo una operazione tecnologica, molto spesso i problemi in un progetto informatico non sono legati alla tecnologia, ma alle persone.

Possono nascere resistenze da parte di utenti 'affezionati' all'idea di fare riferimento a macchine fisiche, e non a server virtuali o a desktop virtuali. La reazione umana ai nuovi sistemi è vista spesso come l'ottavo livello della pila OSI, "il livello politico" che è sempre il più difficile da superare per completare un progetto di trasformazione tecnologica come la virtualizzazione.

Grid Computing

E' un modello di calcolo parallelo e distribuito che può significare:

- La **creazione di un supercalcolatore virtuale** composto da un insieme di sistemi di calcolo collegati in rete che operano di concerto per risolvere problemi di calcolo di grandi dimensioni. Ambito scientifico per la risoluzione di problemi di tipo data intensive
- **Calcolo e spazio di memoria offerto come servizio** da un insieme limitato e controllato di risorse ad un insieme limitato e controllato di utenti. Organizzazioni virtuali.
- I **temi critici** sono quelli dell'autenticazione, dell'autorizzazione e della **sicurezza** che un middleware deve fornire.

The Anatomy of the Grid: Enabling Scalable Virtual Organizations

I. Foster , C. Kesselman, S. Tuecke

International Journal of High Performance Computing Applications

2001

Utility Computing

Utility computing è un modello di business per fornire risorse di calcolo.

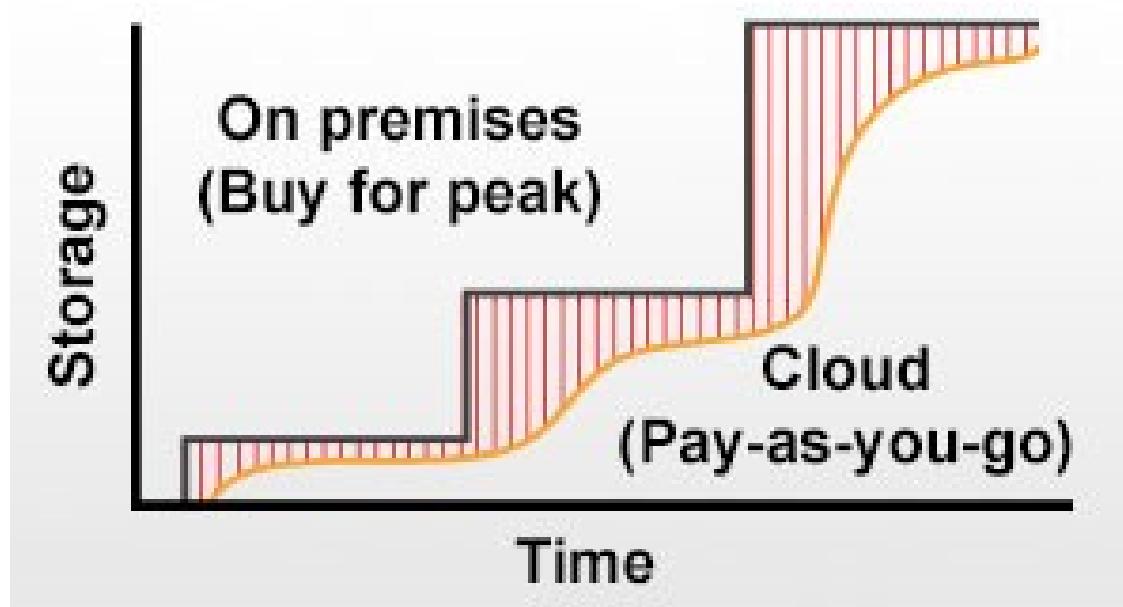
L'utente acquisisce l'uso di una risorsa da un fornitore di servizi e paga per la quantità di risorse (CPU, memoria, disco, ...) utilizzate.

Il vantaggio principale è economico, cioè l'utilizzatore paga solamente quello che usa e quando lo usa.

- I servizi di calcolo centralizzati sono spesso sottoutilizzati (si valuta che i server siano "idle" per l'85% del tempo).
- Si è spesso in presenza di sovrardimensionamento (overprovisioning) dell'hw necessario in media, al fine di poter sopravvivere durante i picchi di richieste (come se si dimensionassero le autostrade per sopportare agevolmente il traffico di ferragosto).

Utility computing nell'IT: Cloud Computing

- Le esigenze di risorse (Storage, CPU) possono essere soddisfatte con l'acquisto (overprovisioning) oppure con l'utilizzo (oneroso) di ciò che occorre



Cloud Computing

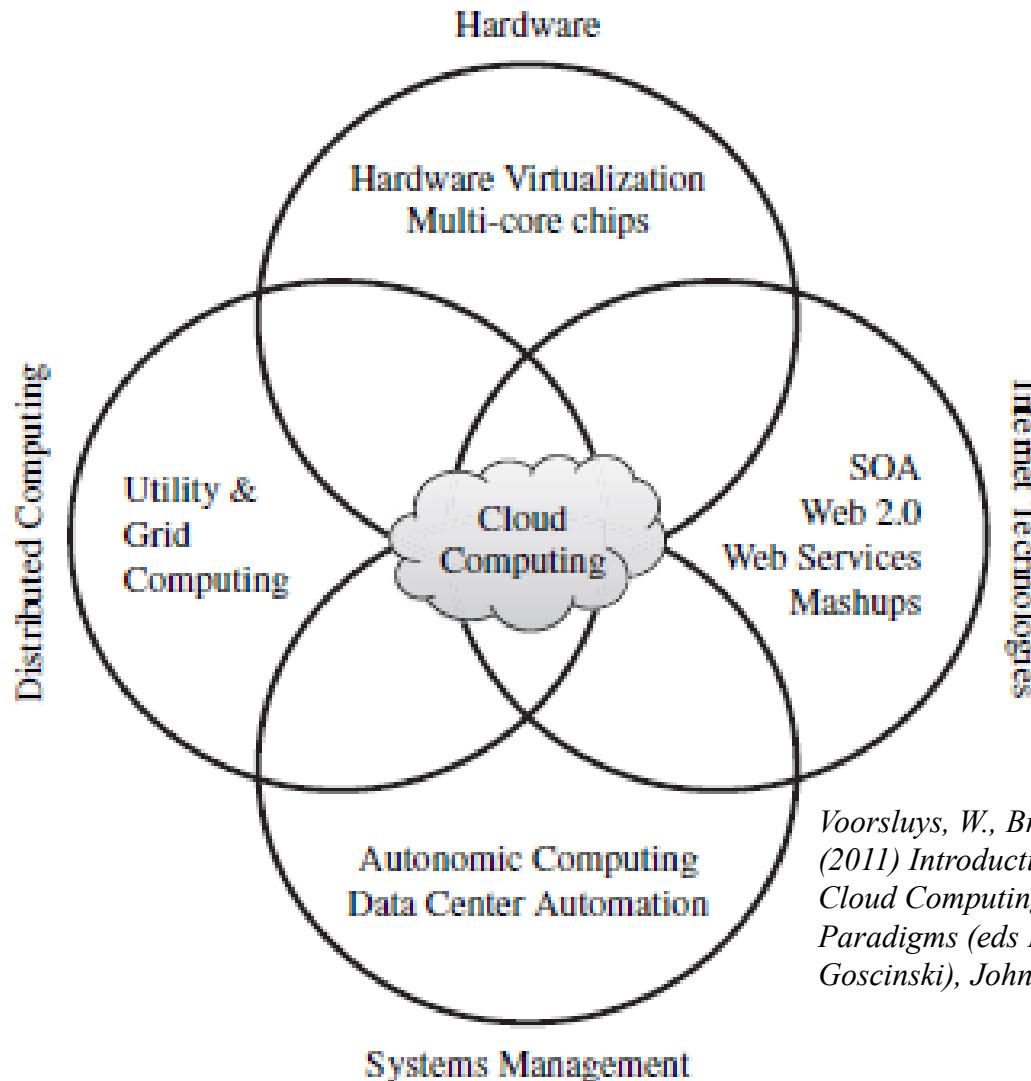
Cloud Computing è il nuovo concetto introdotto nel mercato dell'informatica per identificare il modo con il quale verranno gestiti i servizi informatici in futuro.

Cloud computing è un modo di mettere a disposizione e usare risorse (hardware e software) esclusivamente tramite la rete che promette di metterne a disposizione una quantità che cresce in modo indefinito.

Cloud computing richiede e sfrutta le nuove tecnologie che sono a disposizione e in particolare: i nuovi processori, la tecnologia di virtualizzazione, le nuove architetture di storage distribuito, le reti di accesso ad Internet a larga banda.

Termini che introducono al Cloud computing sono: Grid computing, Utility computing, Virtualizzazione, Cluster di server, Outsourcing, PaaS (Platform as a service), SaaS (Software as a service).

Cloud Computing



Voorsluys, W., Broberg, J. and Buyya, R. (2011) *Introduction to Cloud Computing*, in *Cloud Computing: Principles and Paradigms* (eds R. Buyya, J. Broberg and A. Goscinski), John Wiley & Sons

Come definire il Cloud Computing?

- Buyya et al. : "Cloud is a parallel and distributed computing system consisting of a collection of inter-connected and virtualised computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements (SLA) established through negotiation between the service provider and consumers."
- Vaquero et al. "clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for an optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized Service Level Agreements."
- McKinsey and Co. report : "Clouds are hardware based services offering compute, network, and storage capacity where: Hardware management is highly abstracted from the buyer, buyers incur infrastructure costs as variable OPEX, and infrastructure capacity is highly elastic."
- A report from the University of California Berkeley: "(1) the illusion of infinite computing resources; (2) the elimination of an up-front commitment by cloud users; and (3) the ability to pay for use . . . as needed . . ."
- National Institute of Standards and Technology (NIST) characterizes cloud computing as ". . . a pay-per-use model for enabling available, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."
- Armbrust et al. define cloud as the "data center hardware and software that provide services."

Tipologie di Cloud Computing

BaaS/BPaaS : Business (process) as a Service

SaaS : Software as a Service

DaaS : Data as a Service

PaaS : Platform as a Service

IaaS : Infrastructure as a Service

HaaS : Hardware as a Service

Tipi principali di Cloud

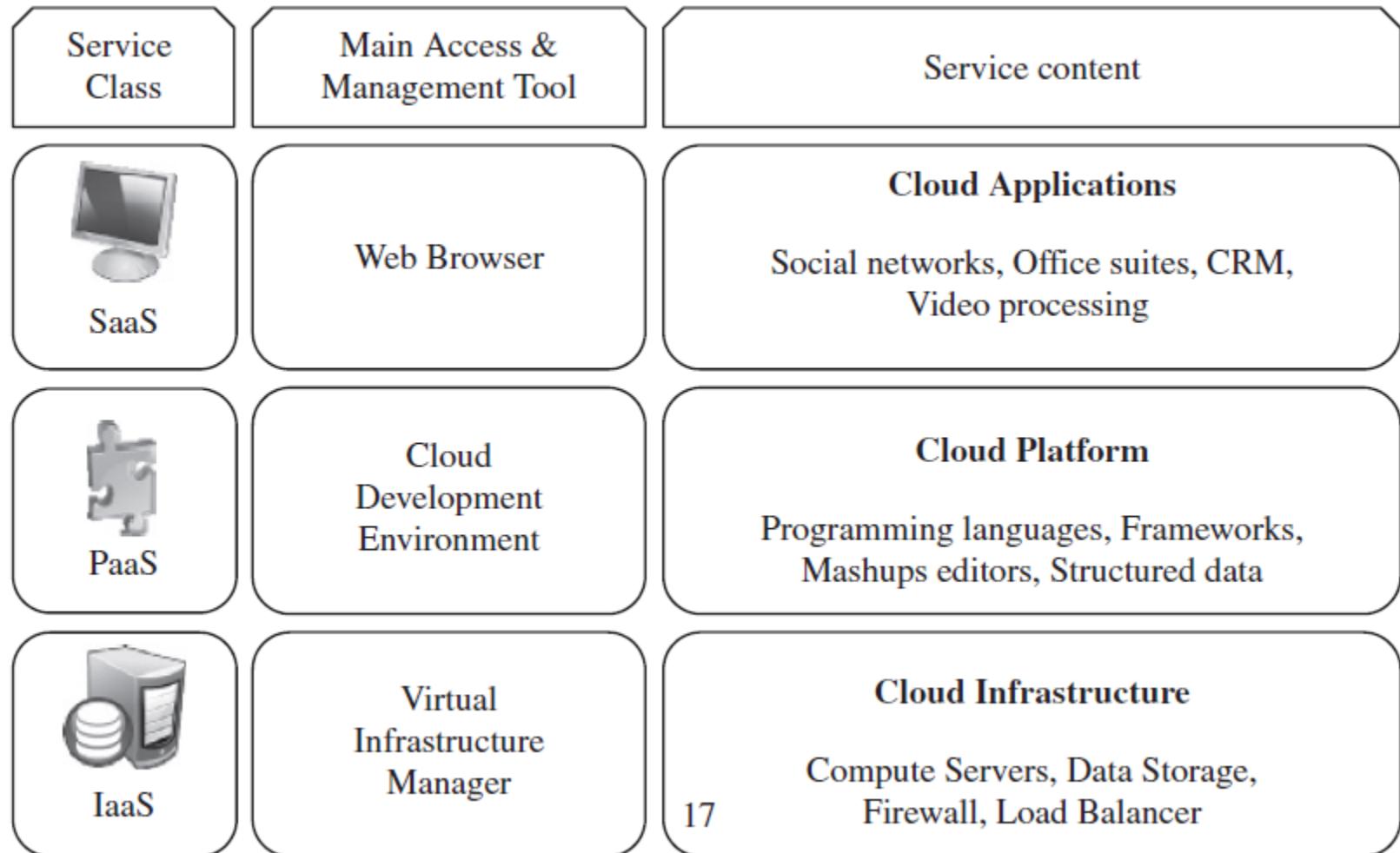
- **Software as a Service (SaaS)**

- Applications reside on the top of the cloud stack.
- Services provided by this layer can be accessed by end users through Web portals
- Consumers are increasingly shifting from locally installed computer programs (WP/SP) to on-line software services that offer the same functionality

NIST definition

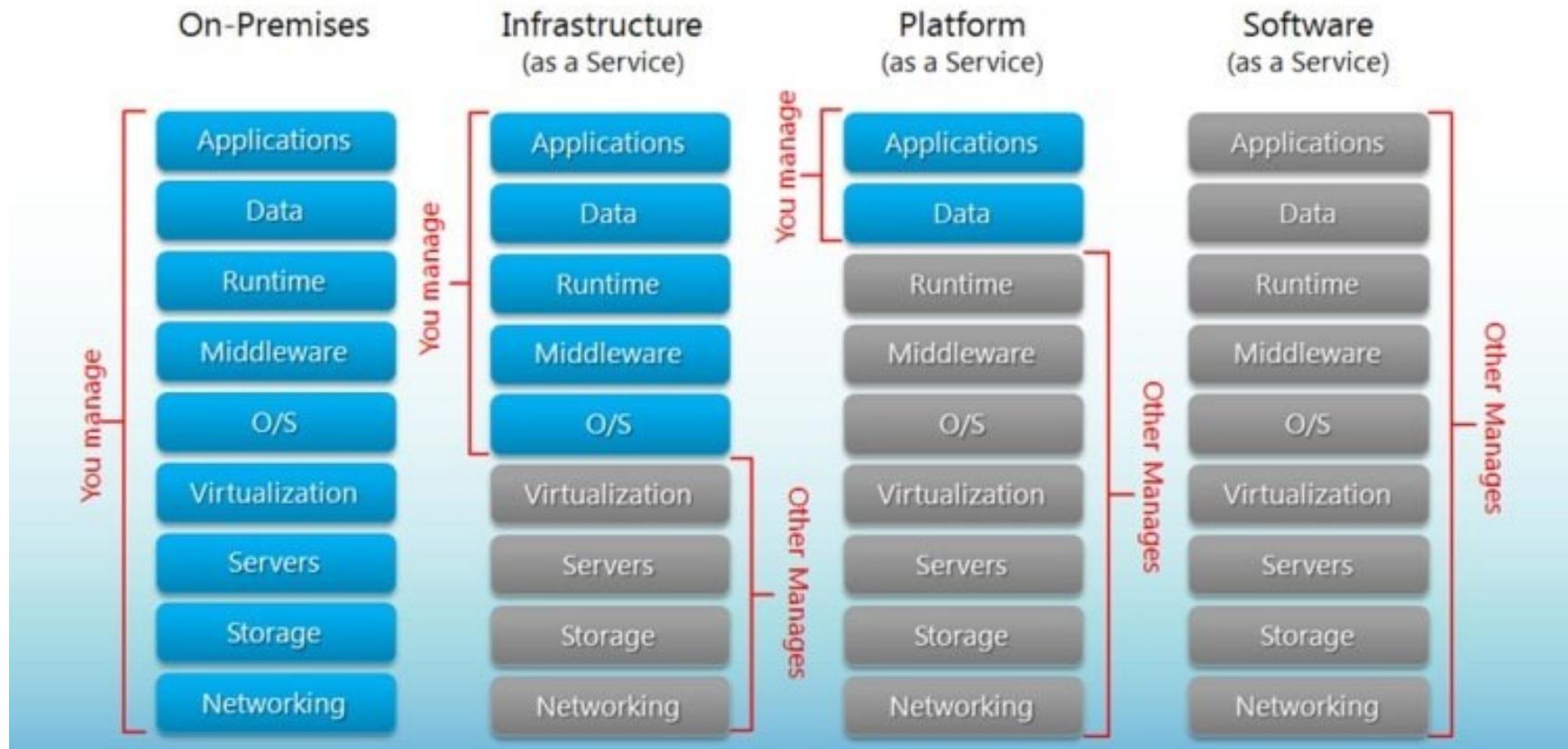
- *The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure*
- *The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface*
- *The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings*

Cloud computing stack



Voorsluys, W., Broberg, J. and Buyya, R. (2011) *Introduction to Cloud Computing*, in *Cloud Computing: Principles and Paradigms* (eds R. Buyya, J. Broberg and A. Goscinski), John Wiley & Sons

Responsabilità della gestione risorse



You → Cliente

Other → Service Provider

Tipi principali di Cloud

- **Platform as a Service (PaaS)**

- a higher level of abstraction to make a cloud easily programmable
- offering an environment on which developers create and deploy applications and do not necessarily need to know how many processors or how much memory that applications will be using
- multiple programming models and specialized services (e.g., data access, authentication, and payments) are offered as building blocks to new applications
- (NIST) *The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.*

Tipi principali di Cloud

- **Infrastructure as a Service (IAAS)**
 - offering virtualized resources (computation, storage, and communication) on demand
 - (NIST) *The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls).*

Altri tipi di Cloud

- **BaaS/BPaaS - Business (Process) as a Service** : Any business process (for example, payroll, printing, ecommerce) delivered as a service over the Internet and accessible by one or more web enabled interfaces (PC, smart devices and phones). Both publicly available and private services (IT departments provide to their users within the firewall or to the trusted partners).
- **DaaS - Data as a Service** : delivering specific, valuable data on demand, such as business data, often to be integrated with other cloud services (e.g. SaaS).
- **HaaS – Hardware as a Service** : similarly to leasing, hardware that belongs to a managed service provider (MSP) is installed at a customer's site and a service level agreement (SLA) defines the responsibilities of both parties. Customer pays monthly fee or usage is included in the setup and management fee.

Esempi di SaaS

Salesforce.com	Salesforce.com has become the poster child of software as a service Customer Relationship Management (CRM) software solutions. Now the technology pioneer continues a push into the platform as a service (PaaS) market with its Force.com application platform.
SAP	Enterprise Resource Planning (ERP) giant SAP is assertively pushing into the software as a service CRM and ERP markets with its Business ByDesign solution. The on-demand ERP system is showing strong global adoption from both customers and partners.
Microsoft Office 365	Signature Microsoft productivity applications such as Word, Excel and PowerPoint are longtime staples of the workplace, but the cloud-based Microsoft Office 365 dramatically expands the Office suite's parameters. Users now may create, edit and share content from any PC, Mac, iOS, Android or Windows device in real-time, connect with colleagues and customers across a range of tools from email to video conferencing and leverage a range of collaborative technologies supporting secure interactions both inside and outside of the organization.
Dropbox	Keep your documents and files at your fingertips across all your devices using Dropbox. Anything added to Dropbox storage automatically shows up across all your desktop and mobile devices, enabling professionals to begin a project on their work PC, make edits on their smartphone during the evening commute home, and add the finishing touches from their home tablet. Then users can invite teammates to access any Dropbox folder or send them specific files and images accessible through password-protected links; there's even a remote wipe option in case of emergency.
Shopify	Setting up an online store yourself can be complicated and take outside expertise from developers and designers to make it work. Shopify saw the opportunity to create a platform where you could build an ecommerce store and start selling products in just a few hours. Setting up a Shopify store is easy and painless, providing the site, the shopping cart, online catalog, and payment integration all in the one platform. It also offers customization options via its library of both free and paid plugins.

Esempi di Paas

AWS Elastic Beanstalk	AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.
Heroku	Heroku (owned by SalesForce) is a platform as a service based on a managed container system, with integrated data services and a powerful ecosystem, for deploying and running modern apps. The Heroku developer experience is an app-centric approach for software delivery, integrated with today's most popular developer tools and workflows.
Windows Azure (mostly used as PaaS)	Microsoft Azure not only offers platform as a service but also software as a service and infrastructure as a service. With Azure, clients can use the services purely on the cloud or it can be combined with any existing applications, data centre or infrastructure you may already have in place. Azure's PaaS was one of the solutions early offering which later included IaaS. Many cloud providers are blurring the lines between PaaS and IaaS. Microsoft Azure is no stranger to this idea of mixing and matching both services.
Google App Engine	Google App Engine is a fully managed, serverless platform for developing and hosting web applications at scale. You can choose from several popular languages, libraries, and frameworks to develop your apps, then let App Engine take care of provisioning servers and scaling your app instances based on demand.

Esempi di IaaS

AWS EC2 (Amazon Elastic Computer Cloud)	AWS is a cloud-based program for building business solutions using integrated web services. While it offers much more, AWS is considered an Infrastructure as a Service (IaaS) solution as it supplies the hardware infrastructure that allows businesses to build their web-based solutions in the Cloud rather than physically in their premises.
IBM Cloud	IBM Cloud (formerly known as IBM Soft Layer) does not offer just IaaS, but a wide range of services and not all of them are cloud based either. It covers both virtual and hardware – based servers, composed of public, private and management networks.
Google Cloud Platform	Google Cloud Platform is a suite of cloud computing services that is run on the same infrastructure used by Google internally for its end-user products. It provides Infrastructure as a Service, Platform as a Service and Serverless Computing. The platform was initially released in 2008. Google Cloud Platform enables users to create business solutions using Google-provided, modular web services. While the platform is an IaaS, it also comes with a managed platform and pre-built web services utilities.

Providers	Cloud Promises							
	Scalability and Automation				Choice and Flexibility			
	Scale Up	Scale Out	APIs	Monitoring	Datacenters	Instance Types	Supported Operating Systems	
Top-of-Mind								
Amazon (EC2)	Yes	Yes	Yes	Extensive	8	17	4	
Rackspace	Yes	Yes	Yes	Extensive	9	7	4	
GoGrid	Yes	Yes	Yes	Poor	5	7	7	
Microsoft	Yes	Yes	Yes	Average	8	5	6	
Terremark	Yes	Yes	Yes	Poor	10	32	6	
AT&T	Yes	Yes	No	Poor	26	Configurable	2	
Google	Yes	Yes	Yes	Poor	11	22	2	
OpSource	Yes	Yes	Yes	Average	5	Configurable	4	
Softlayer	Yes	Yes	Yes	Extensive	7	10	4	
HP	Yes	Yes	Yes	Poor	3	6	4	
Upstarts								
BitRefinery	Yes	Yes	No	Poor	4	Configurable	4	
Lunacloud	Yes	Yes	Yes	Poor	5	Configurable	8	
Nephoscale	No	Yes	Yes	Poor	1	13	4	
Tier3	Yes	Yes	Yes	Poor	9	Configurable	9	

Providers	User Concerns					
	Security Features		Ease of Migration		Reliability	
	Certifications	Protection	Standards	VM Upload	Service Age	SLA
Top-of-Mind						
Amazon (EC2)	Yes	Medium	Proprietary	Yes	5+ Years	99.95%
Rackspace	Yes	Medium	OpenStack	No	5+ Years	100.00%
GoGrid	No	Medium	Proprietary	No	5+ Years	100.00%
Microsoft	Yes	Medium	HyperV	Yes	1 - 2 Years	99.95%
Terremark	Yes	Medium	VMware	Yes	2 - 3 Years	100.00%
AT&T	Yes	Medium	VMware	Yes	4 - 5 Years	??
Google	Yes	Medium	Proprietary	No	1 - 2 Years	99.95%
OpSource	Yes	Medium	VMware	Yes	5+ Years	100.00%
Softlayer	Yes	Medium	OpenStack	No	5+ Years	100.00%
HP	No	Medium	OpenStack	No	1 - 2 Years	99.95%
Upstarts						
BitRefinery	Yes	Poor	VMWare	Yes	2 - 3 Years	100.00%
Lunacloud	No	Poor	Proprietary	No	1 - 2 Years	99.99%
Nephoscale	No	Poor	Proprietary	No	1 - 2 Years	99.95%
Tier3	Yes	Medium	VMWare	No	5+ Years	99.90%

Modelli di deployment

- **Private cloud.** The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.
- **Community cloud.** The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises.
- **Public cloud.** The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider.
- **Hybrid cloud.** The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data

Caratteristiche del Cloud Computing

- **Scalabilità nelle performance**

Allocazione dinamica, continua e garantita da una frazione di core fino core multipli per ogni sistema. Scalabilità degli spazi storage illimitata.

- **Scalabilità nei costi**

Costo del servizio variabile a seconda delle performance e delle risorse richieste.

- **Cloud Self Provisioning e Auto Provisioning**

Configurazione online del sistema da parte dell'utente o automatica. Adeguamento realtime delle performance.

- **Servizi di categoria Enterprise**

Alta affidabilità, Continuità indipendente dall'hardware, Helpdesk di II livello

Da: innovazionedigitale s.r.l.

Caratteristiche essenziali del CC per NIST

- **On-demand self-service.** A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed
- **Broad network access.** Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., smartphones, PCs,...)
- **Resource pooling.** The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. Location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter).
- **Rapid elasticity.** Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.
- **Measured service.** Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.

Cloud e clienti

- Pricing : listini (sul web) prezzi dei servizi

Google vs. AWS On-Demand Pricing

Google Instance Type	CPU Cores	RAM	AWS Instance Type	CPU Cores	RAM	Google New On-Demand (per hour)	AWS On-Demand (per hour)	New Google Price vs. AWS (per hour)	New Google Price vs. AWS (per GB-hour)
n1-standard-1	1	3.75	m3.medium	1	3.75	\$ 0.050	\$ 0.070	-28.57%	-28.57%
n1-standard-2	2	7.5	m3.large	2	7.5	\$ 0.100	\$ 0.140	-28.57%	-28.57%
n1-standard-4	4	15	m3.xlarge	4	15	\$ 0.200	\$ 0.280	-28.57%	-28.57%
n1-standard-8	8	30	m3.2xlarge	8	30	\$ 0.400	\$ 0.560	-28.57%	-28.57%
n1-highmem-2	2	13	r3.large	2	15	\$ 0.126	\$ 0.175	-28.00%	-16.92%
n1-highmem-4	4	26	r3.xlarge	4	30.5	\$ 0.252	\$ 0.350	-28.00%	-15.54%
n1-highmem-8	8	52	r3.2xlarge	8	61	\$ 0.504	\$ 0.700	-28.00%	-15.54%
n1-highcpu-2	2	1.8	c3.large	2	3.75	\$ 0.076	\$ 0.105	-27.62%	+50.79%
n1-highcpu-4	4	3.6	c3.xlarge	4	7.5	\$ 0.152	\$ 0.210	-27.62%	+50.79%
n1-highcpu-8	8	7.2	c3.2xlarge	8	15	\$ 0.304	\$ 0.420	-27.62%	+50.79%
n1-highcpu-16	16	14.4	c3.4xlarge	16	30	\$ 0.608	\$ 0.840	-27.62%	+50.79%

After GCE price cuts of May 18, 2015

Source: RightScale

- Service Level Agreement : strumenti contrattuali con cui si definiscono le metriche di servizio (es. QoS, disponibilità, banda) che devono essere rispettate dal provider nei confronti dei propri clienti/utenti, pena rimborsi accreditati (previsti nella SLA) ... **attenzione alle definizioni di QoS e penali...**

Componenti del SLA per servizi Cloud

- **Service guarantee** specifies the metrics which a provider strives to meet over a service guarantee time period. Failure to achieve those metrics will result in a service credit to the customer. **Availability** (e.g., 99.9%), **response time** (e.g., less than 50ms), **disaster recovery**, and **fault resolution time** (e.g., within one hour of detection) are examples of service guarantees. Some service guarantees can be on a per action basis, such as zeroing out a VM disk when it is deprovisioned.
- **Service guarantee time period** describes the duration over which a service guarantee should be met. The time period can be a billing month or time elapsed since the last claim was filed. The time period can also be small, e.g., one hour. The smaller the time period, the more stringent is the service guarantee.
- **Service guarantee granularity** describes the resource scale on which a provider specifies a service guarantee. For example, the granularity can be on a per service, per data center, per instance, or per transaction basis. Similar to time period, the service guarantee can be stringent if the if the granularity of service guarantee is fine-grained.

Salman A. Baset, Cloud SLAs: present and future, ACM SIGOPS Operating Systems Review, v.46 n.2, July 2012

Componenti del SLA per servizi Cloud

- **Service guarantee exclusions** are the instances that are excluded from service guarantee metric calculations. These exclusions typically include abuse of the system by a customer, or any downtime associated with the scheduled maintenance.
- **Service credit** is the amount credited to the customer or applied towards future payments if the service guarantee is not met. The amount can be a complete or partial credit of the customer payment for the affected service.
- **Service violation measurement and reporting** describes how and who measures and reports the violation of service guarantee, respectively.

Salman A. Baset, Cloud SLAs: present and future, ACM SIGOPS Operating Systems Review, v.46 n.2, July 2012

Disponibilità e dettagli

Availability %	Downtime per year	Downtime per month*	Downtime per week
90% ('one nine')	36.5 days	72 hours	16.8 hours
99% ('two nines')	3.65 days	7.20 hours	1.68 hours
99.5%	1.83 days	3.60 hours	50.4 minutes
99.9% ('three nines')	8.76 hours	43.8 minutes	10.1 minutes
99.95%	4.38 hours	21.56 minutes	5.04 minutes
99.99% ('four nines')	52.56 minutes	4.32 minutes	1.01 minutes
99.999% ('five nines')	5.26 minutes	25.9 seconds	6.05 seconds
99.9999% ('six nines')	31.5 seconds	2.59 seconds	0.605 seconds
99.99999% ('seven nines')	3.15 seconds	0.259 seconds	0.0605 seconds

Source: Wikipedia

Come sono calcolati l'uptime e la disponibilità ?

Cosa dovrebbe esserci presso il cliente per avere la copertura ?

Cosa succede in caso di calo delle prestazioni senza un vero downtime ?

Quali sono le penalità per le violazioni di SLA ?

Cosa occorre fare per richiedere un credito ?

SLA e l'onere sui clienti

Both AWS and HP impose strict guidelines in how users must architect their cloud systems for the SLAs to apply in the case of service disruptions, leading to increased costs for users.

AWS's, for example, requires customers to have their applications run across at least two availability zones (AZ), which are physically separate data centers that host the company's cloud services. Both AZs must be unavailable for the SLA to kick in.

HP's SLA, Leong reports, only applies if customers cannot access any AZs. That means customers have to potentially architect their applications to span three or more AZs, each one imposing additional costs on the business.

<https://www.networkworld.com/article/2161940/gartner--amazon--hp-cloud-slas-are--practically-useless-.html>

I grandi provider di Cloud Computing

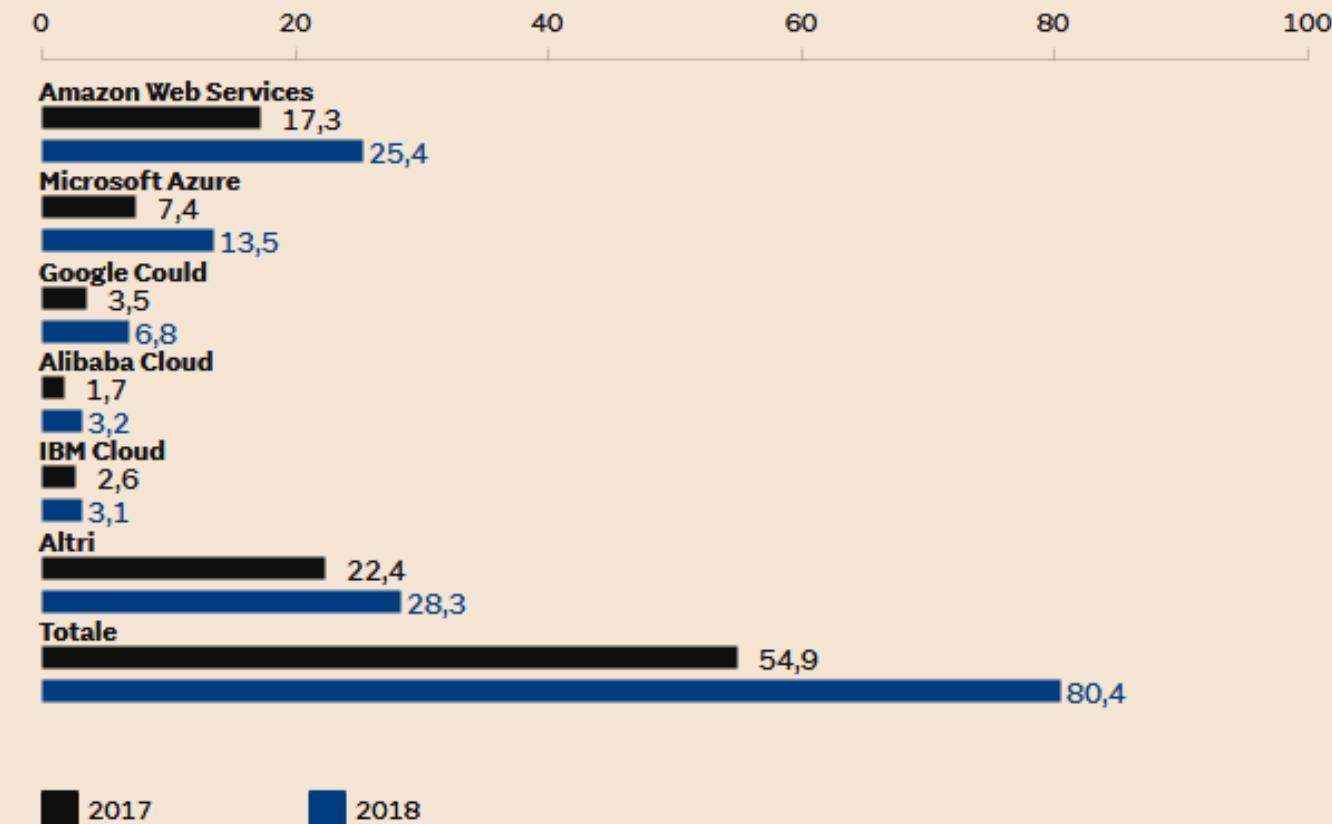
- Amazon AWS (EC2,S3,...)
- Google
- Microsoft Azure
- IBM
- Oracle

E l'Europa?

Il mercato globale del cloud

CHI SONO I BIG DELLA “NUVOLA”

In mld di dollari il mercato globale dei cloud provider. (Fonte: Canalys)



I provider italiani di Cloud Computing

- TIM
- Aruba Cloud
- Seeweb
- FabbricaDigitale
- Enter
- MomIT
- Altri...

R. Stallmann e il Cloud Computing

Richard M. Stallman, (teorico del free software e di GNU):
" il Cloud Computing è roba da stupidi e utilizzare applicazioni web come Gmail di Google è anche peggio della stupidità stessa"

Per Stallman le aziende che hanno investito sulle tecnologie basate sul Cloud Computing hanno in realtà intenzione di chiudere l'utente all'interno di software proprietario per poi costringerlo a pagare sempre di più nel corso degli anni per poter utilizzare i suoi stessi dati e documenti.

Stallman crede fermamente che il processo che sta spingendo gli utenti a muovere i propri dati nella nuvola del Cloud Computing è tutt'altro che inevitabile e che la nostra convinzione del contrario dipende solo dalla massiccia dose di marketing a cui siamo sottoposti.

Evoluzione verso il Cloud Computing

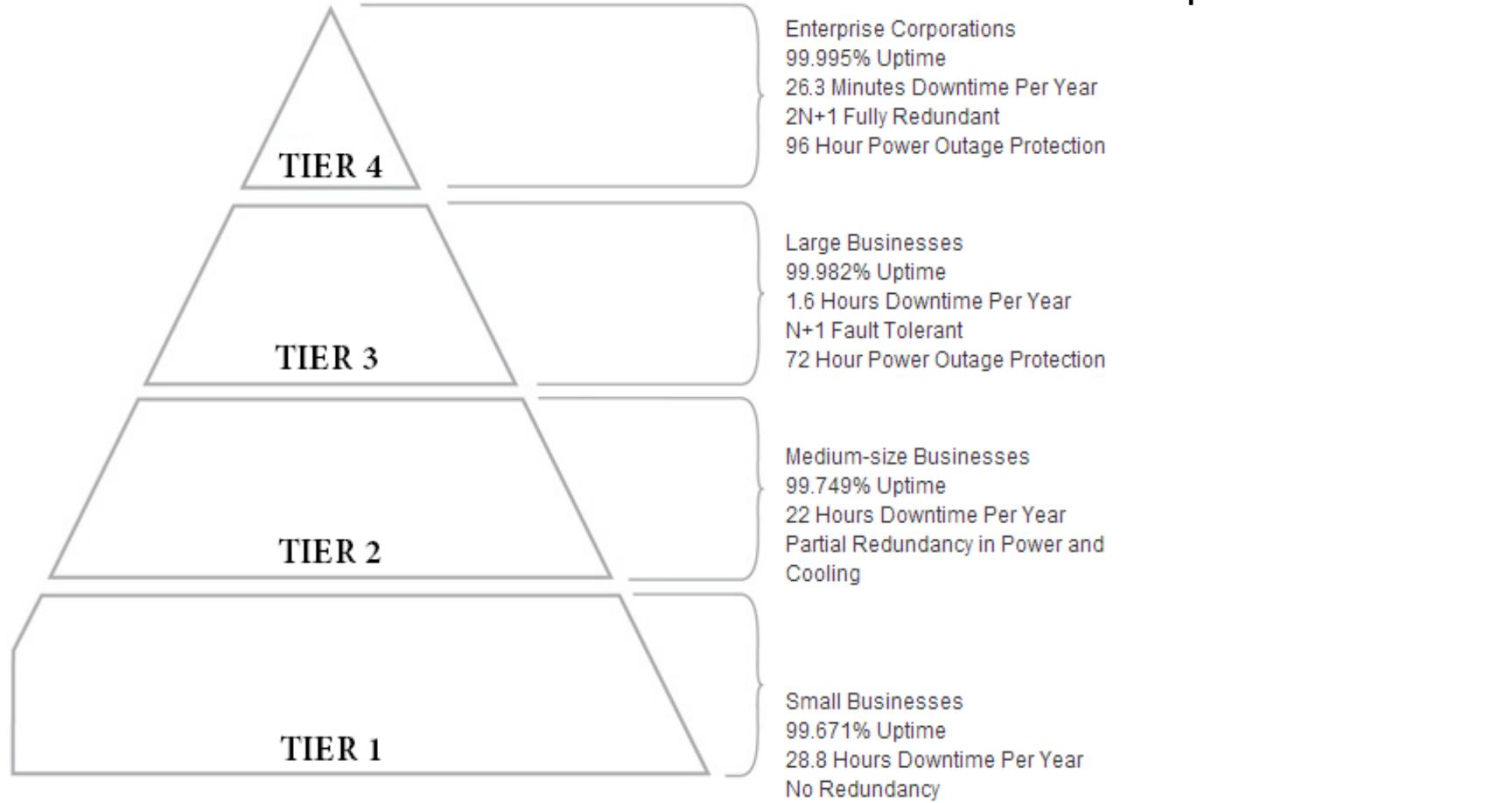
- Passaggio a macchine virtuali di tutti i server
- Creazione di sistemi di server virtuali entro data center propri, organizzati in alta affidabilità (-> Private Cloud)

Evoluzione del Cloud Computing

- Spostamento dei server virtuali entro data center di provider (-> Public Cloud)
- Alta affidabilità e mirror geografico (bilanciamento di carico e fault tolerance)
- Spostamento progressivo degli applicativi da configurazioni IaaS a configurazioni PaaS

Data Center

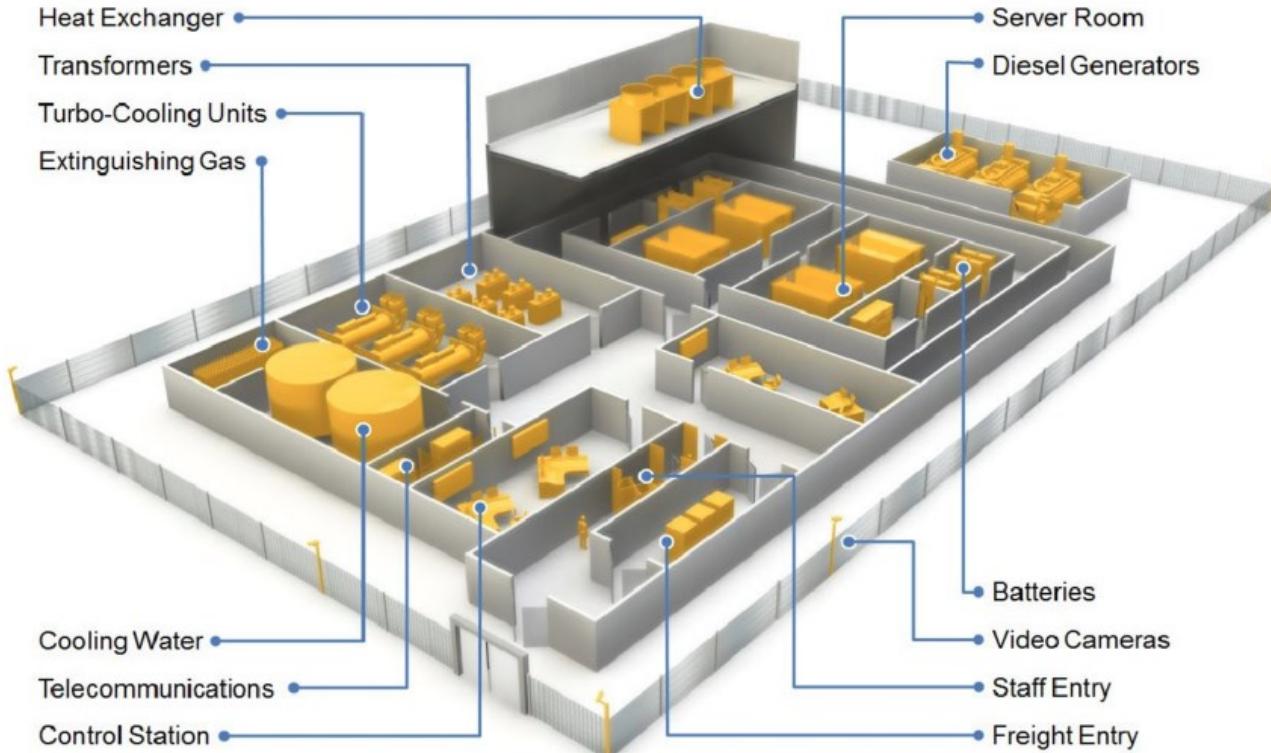
Data Center Tiers



Classificazione
Uptime Institute

Come opera un data center ? http://www.sapdatacenter.com/article/data_center_functionality/#!

Come opera un Data Center?



<https://www.sap.com/about/trust-center/data-center.html>

An Insider's Look: Google's Data Centers (Cloud Next '19)
Inside a Google data center

Data center e efficienza energetica

- Enormi consumi di elettricità
 - Data center Lepida di Parma : Tier III, 800KW consumi di cui 600KW per apparati IT
 - Mega data center : 100 MW di energia utilizzata
- Power Usage Efficiency :
$$\frac{\text{Energia Totale}}{\text{Energia per IT}}$$
 - Data center efficienti : PUE 1.2 o inferiore, anche 1.05 (dipende anche dal clima)
 - Un data center aziendale fatica a scendere sotto PUE = 2
 - Meglio utilizzare risorse ottimizzate e condivise

I Data Center Lepida

- *Lepida SpA realizza e gestisce datacenter regionali: Parma (DUC B), Ferrara (ex Macello), Ravenna (Bassette) con architettura fisica Tier III (livello di ridondanza N+1) prevista da AgID e PUE (Power Usage Effectiveness) tra 1,3 e 1,4 contro la media attuale superiore a 2,5.*
- *Datacenter regionali costruiti come ulteriori 4 POP del core della Rete Lepida, garantendo la massima flessibilità di interconnessione ai Soci, prestazioni elevatissime, alta disponibilità del servizio e livelli di sicurezza tali da fornire il rispetto di tutti i requisiti di privacy richiesti.*
- *Modelli di erogazione in logica cloud: provisioning automatico, elasticità, misurazione delle risorse utilizzate, multitenant con unica infrastruttura condivisa tra più Enti ma con separazione che garantisce la assoluta riservatezza delle informazioni, flessibilità operativa con attivazioni risorse su richiesta e per brevi periodi. Gestione del servizio a carico dell'utilizzatore.*
- *Configurazione di rete e sistematica per operare in mutuo Disaster Recovery.*
- <https://www.lepida.it/datacenter-cloud/virtual-tour>
- https://www.lepida.it/sites/default/files/u8/Listino%20Reti,%20DataCenter&Cloud,%20Servizi%20-%20Listino_3.3.pdf

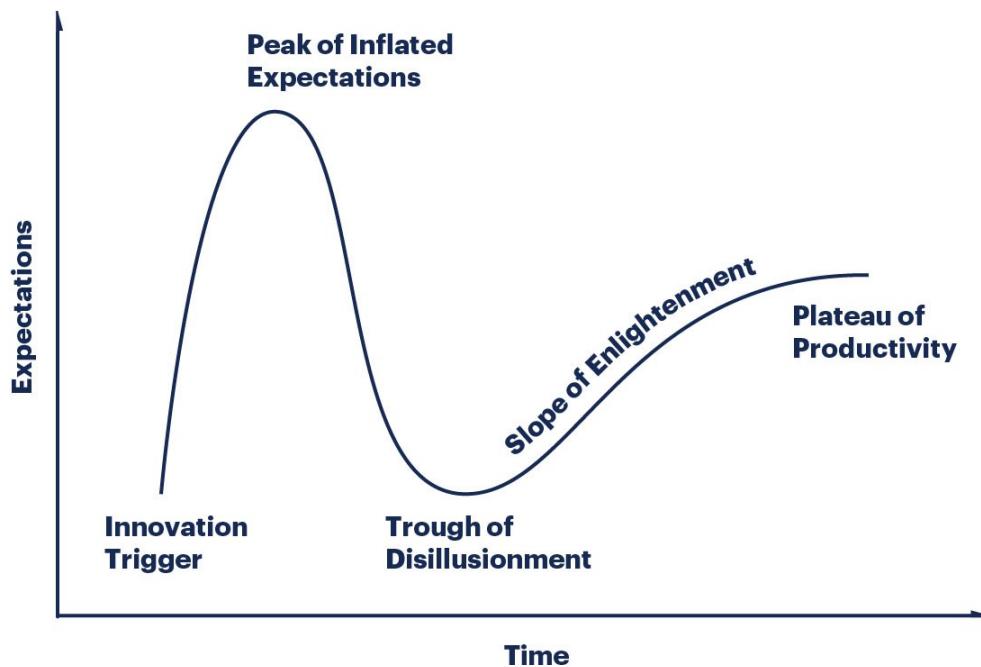
Servizi dei DC Lepida

- Backup as a service Bilanciamento e firewall Disaster Recovery
Database as a service Server virtuale Storage as a service
- Housing Materiali
- *I servizi del DataCenter & Cloud di LepidaSpA permettono agli Enti Soci di fruire di servizi ICT in maniera efficiente secondo un nuovo approccio dinamico, che sfrutta l'evoluzione tecnologica della virtualizzazione dei datacenter di nuova generazione, consentendo alle PA di superare i limiti delle tradizionali infrastrutture ICT on-premise.*
- *Nei DataCenter Cloud di LepidaSpA vengono utilizzate architetture completamente ridondante (dalle facilities di Datacenter fino alle componenti IT) in maniera tale da garantire un ambiente ICT "Enterprise Class".*
- *Il Community cloud di LepidaSpA offre le potenzialità, l'efficienza e le caratteristiche di un public cloud, ma con la sicurezza, il controllo e le prestazioni di un ambiente costruito ad hoc per la PA della CNER.*
- *Tutti i layer dello stack IT-network geografico, network di datacenter, storage, VM, backup, DB - concorrono a definire una soluzione di Virtual datacenter unica e nativamente integrata alla Rete Lepida.*

Innovazione nell'IT

- Velocità negli affari: decisioni tempestive, la velocità è un valore.
- Il mondo è piatto: Risposte in linea in tempo reale fra aziende in rete, approvvigionamento in tutto il mondo (globalizzazione).
- Diminuzione dei costi: esternalizzazione/outsourcing, efficienza energetica.
- Mobilità: consumatori e operatori connessi sempre e ovunque, dispositivi mobili sempre connessi ad internet.
- Green computing: occorre usare le risorse informatiche in modo efficiente: riduzione dei costi, responsabilità sociale, riduzione delle emissioni di CO₂.

Hype cycle di Gartner sul ciclo di vita delle tecnologie



Each Hype Cycle drills down into the five key phases of a technology's life cycle.

Innovation Trigger: A potential technology breakthrough kicks things off. Early proof-of-concept stories and media interest trigger significant publicity. Often no usable products exist and commercial viability is unproven.

Peak of Inflated Expectations: Early publicity produces a number of success stories — often accompanied by scores of failures. Some companies take action; many do not.

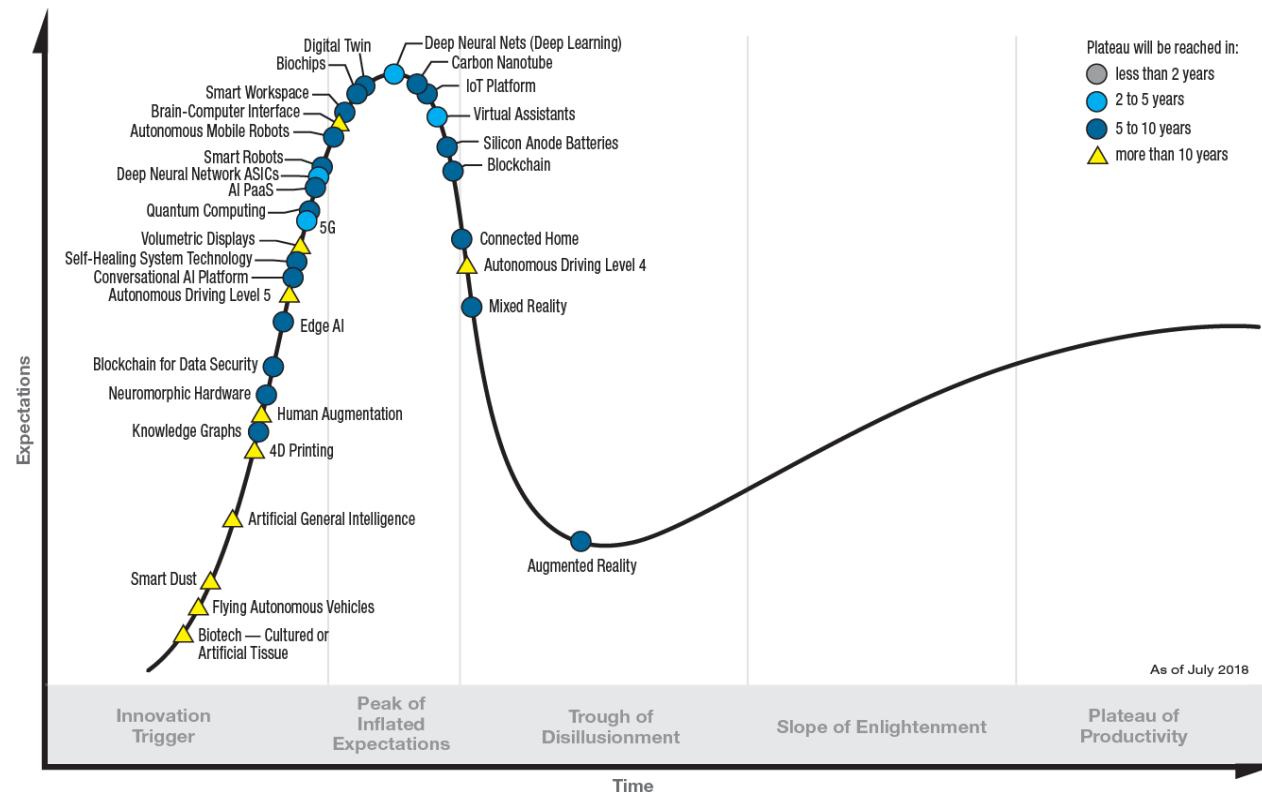
Trough of Disillusionment: Interest wanes as experiments and implementations fail to deliver. Producers of the technology shake out or fail. Investments continue only if the surviving providers improve their products to the satisfaction of early adopters.

Slope of Enlightenment: More instances of how the technology can benefit the enterprise start to crystallize and become more widely understood. Second- and third-generation products appear from technology providers. More enterprises fund pilots; conservative companies remain cautious.

Plateau of Productivity: Mainstream adoption starts to take off. Criteria for assessing provider viability are more clearly defined. The technology's broad market applicability and relevance are clearly paying off.

La visione di Gartner sulle tecnologie emergenti (2018)

Hype Cycle for Emerging Technologies, 2018



gartner.com/SmarterWithGartner

Source: Gartner (August 2018)
© 2018 Gartner, Inc. and/or its affiliates. All rights reserved.

Gartner

La visione di Gartner sulle tecnologie emergenti (2019)

Gartner Hype Cycle for Emerging Technologies, 2019



Emerging Technology Trends 2019



Sensing and Mobility

- 3D sensing cameras
- AR cloud
- Light-cargo delivery drones
- Flying autonomous vehicles
- Autonomous driving Levels 4 and 5



Augmented Human

- Biochips
- Personification
- Augmented intelligence
- Emotion AI
- Immersive workspaces
- Biotech (cultured or artificial tissue)



Postclassical Compute and Comms

- 5G
- Next-generation memory
- Low-earth-orbit satellite systems
- Nanoscale 3D printing



Digital Ecosystems

- DigitalOps
- Knowledge graphs
- Synthetic data
- Decentralized web
- Decentralized autonomous organizations



Advanced AI and Analytics

- Adaptive machine learning (ML)
- Edge AI
- Edge analytics
- Explainable AI
- AI PaaS
- Transfer learning
- Generative adversarial networks
- Graph analytics

gartner.com/SmarterWithGartner

© 2019 Gartner, Inc. All rights reserved. CTMKT_741609

Gartner

gartner.com/SmarterWithGartner

Source: Gartner

© 2019 Gartner, Inc. and/or its affiliates. All rights reserved.

Gartner

La visione di Gartner sulle tecnologie emergenti (2020)

Hype Cycle for Emerging Technologies, 2020

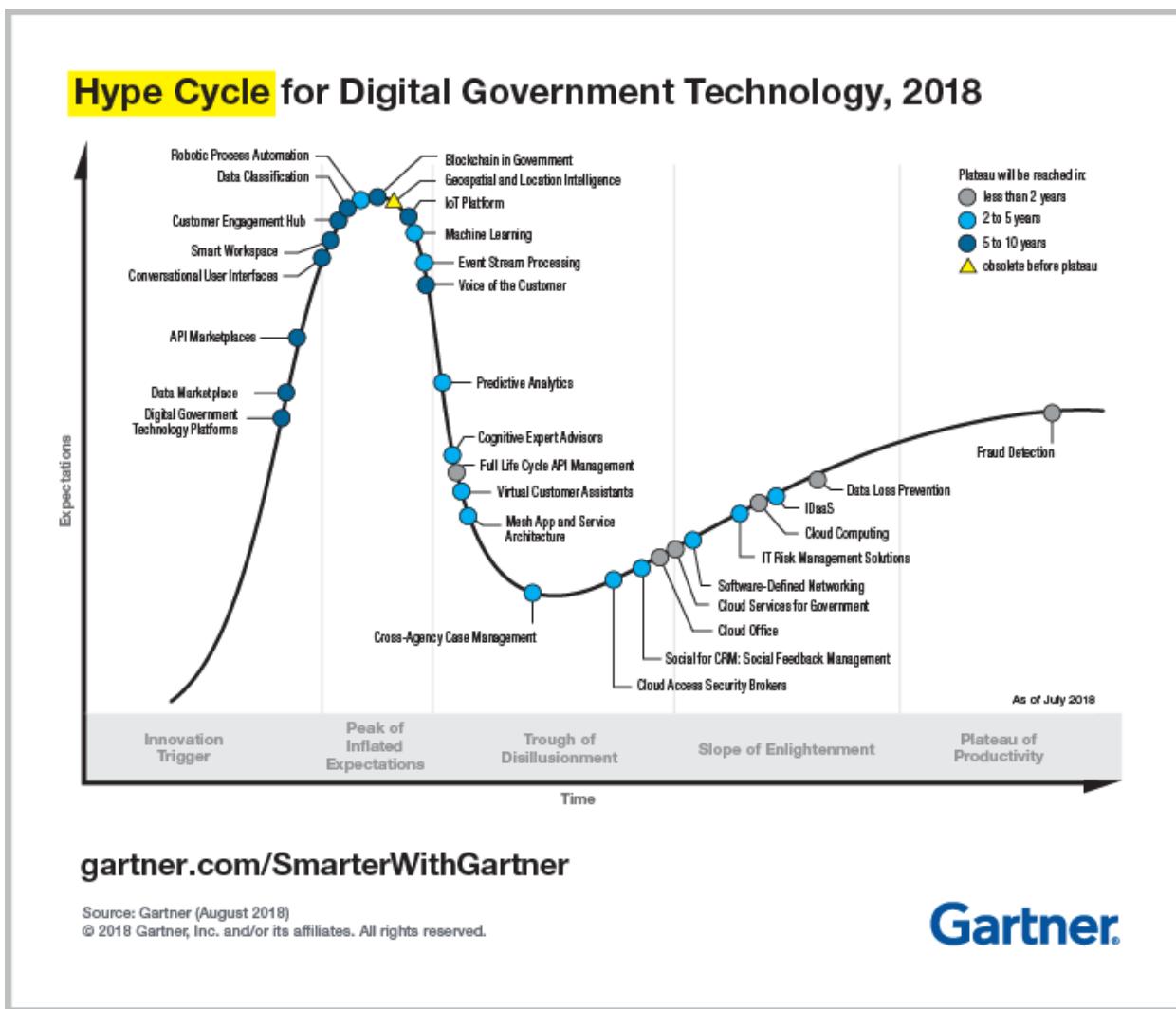


gartner.com/SmarterWithGartner

Source: Gartner
© 2020 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner and Hype Cycle are registered trademarks of Gartner, Inc. and its affiliates in the U.S.

Gartner

La visione di Gartner sulle tecnologie per il Digital Government (2018)



Five technologies for government organizations over the next 10 years.

- **No. 1. Blockchain**

- Blockchain has many potential applications in government. The most transformational impact will be on non-financial uses such as identities, voting, public records and citizen transactions.
 - “Approach blockchain with a healthy dose of skepticism”

- **No. 2. Machine learning**

- Machine learning can solve problems across a vast array of government, citizen and social scenarios. Some examples includes automation, citizen engagement, workforce effectiveness, resource optimization and predictive maintenance.
 - The technology helps governments, and smart cities in particular, progress toward being more anticipatory and pre-emptive.

- **No. 3. Smart workspace**

- Smart workspaces exploit the growing digitalization of physical objects generated by the Internet of Things (IoT). They deliver new ways of working, scheduling resources, coordinating facility services, sharing information and collaborating.

Five technologies for government organizations over the next 10 years.

- **No. 4. Conversational user interfaces**

- A conversational user interface (UI) is a high-level design model in which interactions primarily occur in the user's spoken or written natural language. They make machines smarter and improve people's ability to handle novel situations.
- The promise is a dramatic shift from the user having to learn software, to the interface learning what the user wants – although current UIs aren't yet up to speed.

- **No. 5. Event stream processing**

- Event stream processing (ESP) enables continuous intelligence and other aspects of digital government. It processes data in an event stream, which is a sequence of event objects arranged in order, typically by time.
- The resulting analytics improve decision making by presenting information that could be overlooked. Smarter anomaly detection and faster responses to threats and opportunities are other benefits. They also shield people from data overload by eliminating irrelevant information.
- ESP growth will come from IoT, customer experience management and fraud detection applications.

Fog/Edge Computing

- A horizontal, system-level architecture that distributes computing, storage, control and networking functions closer (edge) to the users along a cloud-to-thing continuum
- Importante per applicazioni IoT (Internet of Things)



IEEE 1934-2018

IEEE Standard for Adoption of OpenFog Reference Architecture for Fog Computing

Fog/Edge Computing

- Gartner defines edge computing as solutions that facilitate data processing at or near the source of data generation
- For example, in the context of the Internet of Things (IoT), the sources of data generation are usually things with sensors or embedded devices. Edge computing serves as the decentralized extension of the campus networks, cellular networks, data center networks or the cloud.

